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/TWC.2019.2940667, IEEE Transactions on Wireless Communications

The Study of Dynamic Caching via State Transition Field - the Case of Time-Varying Popularity

Jie Gao, *Member, IEEE*, Lian Zhao, *Senior Member, IEEE*, and Xuemin (Sherman) Shen, *Fellow, IEEE*

***Abstract*—In the second part of this two-part paper, we extend the study of dynamic caching via state transition field (STF) to the case of time-varying content popularity. The objective of this part is to investigate the impact of time-varying content popularity on the STF and how such impact accumulates to affect the performance of a replacement scheme. Unlike the case in the first part, the STF is no longer static over time, and we introduce instantaneous STF to model it. Moreover, we demonstrate that many metrics, such as instantaneous state caching probability and average cache hit probability over an arbitrary sequence of requests, can be found using the instantaneous STF. As a steady state may not exist under time-varying content popularity, we characterize the performance of replacement schemes based on how the instantaneous STF of a replacement scheme after a content request impacts on its cache hit probability at the next request. From this characterization, insights regarding the relations between the pattern of change in the content popularity, the knowledge of content popularity exploited by the replacement schemes, and the effectiveness of these schemes under time-varying popularity are revealed. In the simulations, different patterns of time-varying popularity, including the shot noise model, are experimented. The effectiveness of example replacement schemes under time-varying popularity is demonstrated, and the numerical results support the observations from the analytic results.**

***Index Terms*—cache replacement policy, content popularity, shot noise model, temporal locality, online caching, mobile edge caching.**

I. Introduction

Driven by the upsurge in the number of user devices and their demand for multimedia services, the role of caching in improving the content delivery performance of wireless networks becomes prominent [1] - [3]. Accordingly, the modeling and analysis of caching have gained tremendous research attention [4]- [7]. While the independence reference model (IRM) is the de facto model for content requests, it has been argued that the IRM may not be sufficiently accurate in practice since temporal correlation of content requests can be too important to neglect [8]. As a result, one particular topic, i.e., online caching with time-varying content popularity, has attracted great research interest lately [9], [10].

The above-mentioned temporal correlation of content requests is sometimes referred to as 'temporal locality', which suggests that a recently requested content is likely to be requested again in the near future. Temporal locality, however, has been shown to emerge from the temporal correlation of requests, the content popularity, or both [11]. Therefore, temporal locality exists even with IRM, and time-varying content popularity complicates the locality by introducing the temporal correlation. As a result, the study of online caching in the case of time-varying content popularity can be very challenging [9]. Existing research on caching with time-varying content popularity can be roughly categorized into two groups: the first group of works aims to analyze or model temporal locality, and the second group targets at proposing caching solutions to cope with it.

Some early works on analyzing temporal locality focused on understanding its sources and developing metrics to measure it, e.g., [12]. In a recent work, Zhou *et al.* investigated the change of popularity over time in the video-on-demand services [13]. While the above studies tend to be experiment-based, mathematical models for characterizing temporal locality can be found in a few works. An inter-reference gap model was developed in [14], which focused on describing temporal locality based on the gaps between successive requests. Traverso *et al.* proposed a shot noise model [15], which represents the requests for a content with an inhomogeneous Poisson process, and later applied it on the analysis of video-on-demand traffic [16]. Other approaches to integrate temporal locality into the analysis of caching also exist, most of which modeled the request for each content as a (semi-)Markov-modulated process or a renewal process [17], [18].

By comparison, a larger number of works can be found in the second group, which proposes caching solutions to cope with temporal locality. Such solutions generally require the prediction of locality or the learning of content popularity. A cache replacement scheme based on predicting the interval between requests was proposed in [19] and shown to be effective in increasing cache hits. Li *et al.* developed a popularity-driven cache replacement scheme which learns the content popularity in an online fashion and makes replacement decisions based on the popularity forecast [20]. Zhang *et al.* proposed a model-free reinforcement learning algorithm for cache replacement based on a linear content popularity prediction model [21]. The above works can be labeled as online caching based on learning/prediction since decisions for cache update are made after every content request. Another type of solutions is proactive caching based on prediction, which can handle time-varying content popularity assuming that cached contents are updated with a sufficiently high frequency. Sadeghi *et al.* exploited reinforcement learning to track content popularity in an online fashion and developed a Q-learning based algorithm for content placement [22]. Applegate *et al.* formulated content placement as an optimization problem and, through estimating content popularity, proposed strategies to update

J. Gao and X. Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, N2L 3G1, Canada (e-mail: {jie.gao, sshen}@uwaterloo.ca).

L. Zhao is with the Department of Electrical, Computer, and Biomedical Engineering, Ryerson University, Toronto, ON, M5B 2K3, Canada (e-mail: l5zhao@ryerson.ca).

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications

cache contents to track time-varying content popularity [23]. Bharath *et al.* characterized the performance of caching with non-stationary content popularity from a learning-theoretic perspective and proposed a cache update policy based on the estimation of content popularity [24].

Evidently, understanding the impact of time-varying content popularity on the performance of caching is important for the analysis and design of cache replacement schemes. However, analysis regarding the impact of time-varying popularity on the performance of replacement schemes is limited in the existing literature. The state transition field (STF) that we proposed in [25] can be used for such analysis. However, with time-varying content popularity, the STF is no longer a static field but a dynamically varying field, and, consequently, a steady state may not exist. The objective of the second part of this two-part paper is to investigate the impact of time-varying content popularity on the STF and, as a result, the performance of replacement schemes.

The contributions of the second part are the followings.

First, we extend the concept of STF from the first part of this two-part paper [25] and introduce instantaneous STF to characterize replacement schemes in the case of time-varying content popularity. It is shown that many metrics, such as instantaneous state caching probability (SCP) at an arbitrary instant and average cache hit probability over an arbitrary sequence of requests, can be found based on instantaneous STF. The results demonstrate the importance of instantaneous STF in modeling and analyzing replacement schemes with time-varying content popularity.

Second, as steady states may not exist, we characterize performance of a replacement scheme by analyzing the difference in instantaneous cache hit probability with and without applying that scheme after a content request. The result reveals insights regarding the relation between the change pattern in content popularity and the effectiveness of replacement schemes. We illustrate the results in the vector space of SCPs and relate them to the knowledge of content popularity exploited by replacement schemes.

Third, we demonstrate instantaneous STF and average cache hit ratio under time-varying popularity with extensive simulations using example schemes. For instantaneous STF, we illustrate its relation with instantaneous content popularity and instantaneous cache hit probability. For average cache hit ratio, we adopt different models of time-varying content popularity, including the shot noise model, and compare the performance of the example schemes. The results verify the observations from analysis and provide guidelines for designing replacement schemes under time-varying content popularity.

II. System Model under Time-Varying Content Popularity

For the sake of presentation clarity, we reintroduce some formulations from the first part of this two-part paper in Sections II and III. The basic system model follows from the basic model in the first part [25]. As the content popularity becomes time-varying, the symbols used here can be categorized into three groups based on their dependence on the time instant of content request or replacement:

- G-1: independent in both [25] and this paper;
- G-2: independent in [25] but dependent in this paper;
- G-3: dependent in [25] and temporal locality introduces further dependence on the time instant in this paper;

A superscript $(\cdot)^{(n)}$ is added on symbols in groups G-2 and G-3 to denote the time instant related to the nth content request or replacement.

A. Request-independent Symbols

Cache State Vector/Matrix: the cache state vector $\mathbf{s}_k$ for state k and the cache state matrix $\mathbf{C}_\mathrm{s} = [\mathbf{s}_1, \ldots, \mathbf{s}_{N_\mathrm{s}}]$, where N_s is the number of cache states.

Neighboring States: the set of neighbors $\mathcal{H}_k$ and the set of content-l neighbors $\mathcal{H}_{k,l}$ of state k, for any k and any $l \notin \mathcal{C}_k$, where $\mathcal{C}_k$ is the set of cached contents in state k.

The above symbols are in group G-1.

B. Request-dependent Symbols

Content Request Probabilities: the probability of content l being requested at request instant n, denoted by $\upsilon_l^{(n)}$, and the overall content popularity at the nth content request, denoted by $\boldsymbol{\upsilon}^{(n)}$. The content request probabilities are in symbol group G-2.

Instantaneous Cache Hit Probability: the instantaneous cache hit probability at the $(n+1)$th request, denoted by $\gamma^{(n+1)}$ is given by:

$$\gamma^{(n+1)} = \left(\boldsymbol{\upsilon}^{(n+1)}\right)^{\mathrm{T}} \boldsymbol{\lambda}^{(n)}, \tag{1}$$

where $\cdot^{\mathrm{T}}$ represents transpose, and $\boldsymbol{\lambda}^{(n)}$ is the content caching probability (CCP) vector after the nth round of request and replacement. It can be seen that $\gamma^{(n+1)}$ is in symbol group G-3. Note that, in a practical network, there can be different metrics for content delivery, e.g., latency. However, the cache hit probability is an underlying factor which other metrics are dependent on. Consequently, improving the cache hit probability can improve the performance under other metrics. For example, if the cache hit probability at an edge server increases, then the need for retrieving contents from the cloud, and thus the average content delivery latency, reduces. Therefore, our study centers around the cache hit probability.

Station Transition Matrices: The conditional state transition matrix and the state transition matrix are generally time dependent and thus denoted by $\boldsymbol{\Theta}_l^{(n)}$ and $\boldsymbol{\Theta}^{(n)}$, respectively, under time-varying content popularity. However, the situation is complicated by the possible choices of various replacement schemes and will be analyzed in details in Section III.

It is worth noting that the relation between state and content caching probabilities from the first part of this two-part paper [25], i.e.,

$$\boldsymbol{\lambda}^{(n)} = \mathbf{C}_\mathrm{s} \boldsymbol{\eta}^{(n)}, \tag{2}$$

still applies in the second part, where $\boldsymbol{\eta}^{(n)}$ is the SCP vector after the nth round of request and replacement. The above equation can be rewritten as:

$$\boldsymbol{\eta}^{(n)} = \mathbf{C}_\mathrm{s}^{\mathrm{T}} \left(\mathbf{C}_\mathrm{s} \mathbf{C}_\mathrm{s}^{\mathrm{T}}\right)^{-1} \boldsymbol{\lambda}^{(n)} + \mathbf{n}_C^{(n)}, \tag{3}$$

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications

where $\mathbf{n}_C^{(n)}$ can be any vector in the null space of $\mathbf{C}_\mathrm{s}$ that renders $\boldsymbol{\eta}^{(n)}$ a valid probability vector, i.e., $\boldsymbol{\eta}^{(n)} \succeq 0, \boldsymbol{\eta}^{(n)} \preceq 1$, and $\mathbf{1}^\mathrm{T}\boldsymbol{\eta}^{(n)} = 1$. Therefore, the value of $\mathbf{n}_C^{(n)}$ is dependent on the value of $\boldsymbol{\lambda}^{(n)}$.

The content and state caching probabilities $\boldsymbol{\lambda}^{(n)}$ and $\boldsymbol{\eta}^{(n)}$ belong to group G-3 and will be analyzed in details in Section IV.

III. General Replacement Model and Specific Cases

In this section, the state transition probability matrix of the general replacement model is formulated, followed by the study of the four example replacement schemes introduced in the first part of this two-part paper, i.e., random replacement (RR), replace less popular (LP), replace the least popular (TLP), and least-recently-used (LRU) [25]. Based on the state transition probability matrices, the instantaneous STF is defined at the end of this section.

A. General Replacement Model

Similar to the case in the first part, the state transition probability matrix in the general model can be written as:

$$\boldsymbol{\Theta}^{(n)} = \sum_{l\in\mathcal{C}} \upsilon_l^{(n)} \boldsymbol{\Theta}_l^{(n)}. \tag{4}$$

where $\mathcal{C}$ is the set of all contents, and the conditional cache state transition probability matrix given that content $l \notin \mathcal{C}_k$ is requested, i.e., $\boldsymbol{\Theta}_l^{(n)}$, is given by:

$$\boldsymbol{\Theta}_l^{(n)}(m,k) = \begin{cases} 1, & \text{if } k=m \text{ and } l\in\mathcal{C}_k, \\ 1-\sum\limits_{m\in\mathcal{H}_{k,l}} \phi_{l,e(k,m),k}, & \text{if } k=m \text{ and } l\notin\mathcal{C}_k, \\ \phi_{l,e(k,m),k}, & \text{if } m\in\mathcal{H}_{k,l}, \\ 0, & \text{otherwise}, \end{cases} \tag{5}$$

where $\phi_{l,q,k}$ denotes the probability of replacing content q with content l given that the cache is at state k and content l is requested. Unlike the case with time-invariant content popularity, the conditional cache state transition probability matrix $\boldsymbol{\Theta}_l^{(n)}$ can be implicitly request-dependent as a result of $\phi_{l,q,k}$ being request-dependent. Consider the situation when $e(k,m)=q$ and content q is less popular at instant n but more popular at instant n' compared to content l, i.e., $\upsilon_q^{(n)} < \upsilon_l^{(n)}$ and $\upsilon_q^{(n')} > \upsilon_l^{(n')}$. Consequently, $\phi_{l,q,k}$ can be different at instants n and n' if LRU, LP, or TLP is used, and thus $\boldsymbol{\Theta}_l^{(n)}(m,k)$ can be different from $\boldsymbol{\Theta}_l^{(n')}(m,k)$. Using LRU as an example, the probability of content q being the LRU content can be different at instants n and n'. Therefore, $\boldsymbol{\Theta}_l^{(n)}(m,k)$ is implicitly request-dependent although the request index $\cdot^{(n)}$ does not appear in the right-hand side of eq. (5).

B. RR

It is straightforward to see that the conditional cache state transition probability matrix $\boldsymbol{\Theta}_l(m,k)$ in the case of RR is request-independent and remains the same as that in the first part. The overall state transition probability matrix $\boldsymbol{\Theta}_\mathrm{RR}$, however, becomes dependent on (n) through $\boldsymbol{\upsilon}^{(n)}$:

$$\boldsymbol{\Theta}_\mathrm{RR}^{(n)}(m,k) = \begin{cases} 1-L\phi\sum\limits_{l\notin\mathcal{C}_k}\upsilon_l^{(n)}, & \text{if } k=m, \\ \phi\upsilon_{e(m,k)}^{(n)}, & \text{if } m\in\mathcal{H}_k, \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$

where $\phi \in (0, 1/L]$ represents the conditional replacement probability that any specified cached content is replaced given that the requested content is not in the cache.

C. LP

In LP, an existing content may be replaced by the new content after the nth request if the new content is more likely to be requested at the $(n+1)$th request. The case of LP can be complicated as it involves the prediction of content popularity. Denote the prediction of content popularity at the $(n+1)$th request as $\tilde{\boldsymbol{\upsilon}}^{(n+1)}$. Sort the states in a non-decreasing order based on the sum of predicted request probability of the cached contents, i.e.,

$$\sum_{q\in\mathcal{C}_m} \tilde{\upsilon}_q^{(n+1)} \geq \sum_{q\in\mathcal{C}_k} \tilde{\upsilon}_q^{(n+1)}, \quad \text{if } m\geq k. \tag{7}$$

The state transition probability matrix of LP is then given by:

$$\boldsymbol{\Theta}_\mathrm{LP}^{(n)}(m,k) = \begin{cases} \sum\limits_{q\in\mathcal{C}_k}\upsilon_q^{(n)} + \sum\limits_{l\in\tilde{\mathcal{C}}_{\bar{k}\downarrow}}\upsilon_l^{(n)} + \sum\limits_{l\in\tilde{\mathcal{C}}_{\bar{k}\uparrow}}\upsilon_l^{(n)}(1-\alpha), & \text{if } m=k, \\ \alpha\upsilon_{e(m,k)}^{(n)}\phi_{e(m,k),e(k,m),k}^{(n)}, & \text{if } m>k \text{ and } m\in\mathcal{H}_k, \\ 0, & \text{otherwise}, \end{cases} \tag{8}$$

in which α is the parameter for controlling the replacement probability,

$$\phi_{l,q,k}^{(n)} = \frac{\tilde{\upsilon}_l^{(n+1)} - \tilde{\upsilon}_q^{(n+1)}}{\sum\limits_{\{t|t\in\mathcal{C}_k, \tilde{\upsilon}_t^{(n+1)} < \tilde{\upsilon}_l^{(n+1)}\}} (\tilde{\upsilon}_l^{(n+1)} - \tilde{\upsilon}_t^{(n+1)})}, \tag{9}$$

and

$$\tilde{\mathcal{C}}_{\bar{k}\downarrow} = \left\{ l \mid l\notin\mathcal{C}_k, \tilde{\upsilon}_l^{(n+1)} \leq \min_{t\in\mathcal{C}_k}\{\tilde{\upsilon}_t^{(n+1)}\} \right\}, \tag{10a}$$

$$\tilde{\mathcal{C}}_{\bar{k}\uparrow} = \left\{ l \mid l\notin\mathcal{C}_k, \tilde{\upsilon}_l^{(n+1)} \geq \min_{t\in\mathcal{C}_k}\{\tilde{\upsilon}_t^{(n+1)}\} \right\}. \tag{10b}$$

Note that the prediction $\tilde{\boldsymbol{\upsilon}}^{(n+1)}$ is not necessarily updated for each content request, and, as a result, $\tilde{\boldsymbol{\upsilon}}^{(n+1)}$ can be a constant for a number of requests. The above state transition probability matrix applies regardless of what the predicted popularity stands for (i.e., the prediction can be for the next request or for a time period over multiple requests, etc.).

D. TLP

In TLP, an existing content is replaced after the nth request if it is both: i) the least likely to be requested among the cached content at the $(n+1)$th request; and ii) less likely to be requested at the $(n+1)$th request compared to the new content at the nth request. Sort the states in a non-decreasing

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications

order based on the sum of predicted request probability of the cached contents. The state transition probability matrix of TLP is given by:

$$\boldsymbol{\Theta}_{\mathrm{TLP}}^{(n)}(m,k) = \begin{cases} \sum\limits_{q\in\mathcal{C}_k} \upsilon_q^{(n)} + \sum\limits_{l\in\tilde{\mathcal{C}}_{\bar{k}\downarrow}} \upsilon_l^{(n)} + \sum\limits_{l\in\tilde{\mathcal{C}}_{\bar{k}\uparrow}} \upsilon_l^{(n)}(1-\phi_{l,q^\dagger(k),k}), & \text{if } m=k, \\ \upsilon_{e(m,k)}\phi_{e(m,k),q^\dagger(k),k}^{(n)}, & \text{if } m>k \text{ and } k\in\mathcal{H}_{m,q^\dagger(k)}, \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

where $\phi_{l,q^\dagger(k),k}^{(n)}$ is the conditional probability of replacing $q^\dagger(k)$ with l in state k, and

$$q^\dagger(k) = \underset{t\in\mathcal{C}_k}{\operatorname{argmin}}\{\tilde{\upsilon}_t^{(n+1)}\}. \tag{12}$$

Note that $q^\dagger(k)$ changes over time although the superscript $\cdot^{(n)}$ is neglected here for simplicity of denotations. The value of $\phi_{e(m,k),q^\dagger(k),k}^{(n)}$, where $m>k$ and $k\in\mathcal{H}_{m,q^\dagger(k)}$, can be either 1 or $\tilde{\upsilon}_{e(m,k)}^{(n+1)} - \tilde{\upsilon}_{q^\dagger(k)}^{(n+1)}$, referred to TLP-A (always replace) and TLP-P (probabilistically replace), respectively.

Similar to the case in the first part, $\boldsymbol{\Theta}_{\mathrm{LP}}^{(n)}$ and $\boldsymbol{\Theta}_{\mathrm{TLP}}^{(n)}$ are both lower-triangular matrices.

The relation among the content popularity, the prediction, and the SCP, all of which are time varying, can be very complicated. As our focus is on understanding the impact of replacement schemes on the time-varying SCP instead of predicting content popularity, the prediction in the case of LP and TLP will be assumed to be accurate in this work. Same as in the first part, LP and TLP, unlike RR and LRU, are not practical replacement schemes but considered here just for analyzing the impact of content popularity information on the STF of replacement schemes.

E. LRU

To fit the LRU into the general cache state transition model, the conditional probability that a specific cached content is the LRU given the current cache state needs to be found. In order to find this conditional probability, the following result is obtained.

Lemma 1: The joint probability that: i) the current state is k; ii) content $q^\star\in\mathcal{C}_k$ is the LRU content at the nth request; and iii) the most recent request for $q^\star$ is the $(n-w)$th request, denoted by $\rho^{(n)}(q^\star,w,k)$, can be found by:

$$\rho^{(n)}(q^\star,w,k) = \sum_{u=1}^{U_w}\prod_{i=1}^{L-1}\prod_{t\in\mathcal{T}(k,i,u,\bar{q}^\star)} \upsilon_{k(i,\bar{q}^\star)}^{(t)}. \tag{13}$$

where $k(i,\bar{q}^\star), i\in\{1,\ldots,L-1\}$ represents the ith cached content in state k that is not content $q^\star$, U_w represents the number of all possible ways for ordering and allocating $w-1$ requests to $L-1$ contents while guaranteeing at least one request for each content, and $\mathcal{T}(k,i,u,\bar{q}^\star)$ represents the set of requests allocated to content $k(i,\bar{q}^\star)$ in the uth out of the U_w allocations.

Proof: See Section A in Appendix.

Given the joint probability in Lemma 1, the conditional probability that content $q^\star\in\mathcal{C}_k$ is the LRU content given that the current state is k can be found as follows [1]:

$$\rho^{(n)}(q^\star|k) = \frac{\sum\limits_{w=L}^{\infty}\rho^{(n)}(q^\star,w,k)}{\sum\limits_{w=L}^{\infty}\sum\limits_{q\in\mathcal{C}_k}\rho^{(n)}(q,w,k)}. \tag{14}$$

Note that the above probability is the general case for the probability $\rho_{e(k,m)|k}^{\mathrm{LRU}}$ from the first part of this two-part paper.

Using the above conditional probability, the conditional state transition probability matrix $\boldsymbol{\Theta}_l$ can be given by:

$$\boldsymbol{\Theta}_{l,\mathrm{LRU}}^{(n)}(m,k) = \begin{cases} 1, & \text{if } l\in\mathcal{C}_k \text{ and } k=m, \\ \rho^{(n)}(e(k,m)|k), & \text{if } m=\mathcal{H}_{k,l}, \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

The overall state transition probability matrix $\boldsymbol{\Theta}_{\mathrm{LRU}}$ is given by:

$$\boldsymbol{\Theta}_{\mathrm{LRU}}^{(n)}(m,k) = \begin{cases} \sum\limits_{l\in\mathcal{C}_k}\upsilon_l^{(n)}, & \text{if } k=m, \\ \upsilon_{e(m,k)}^{(n)}\rho^{(n)}(e(k,m)|k), & \text{if } m\in\mathcal{H}_k, \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

IV. Instantaneous CCP and STF

Based on the state transition probability matrix, this section analyzes the transition of the instantaneous CCP and formulates the instantaneous STF.

A. Instantaneous CCP

Based on the relation between the content and the state caching probabilities in eq. (2), the resulting CCP vector after the nth request and replacement is given by:

$$\boldsymbol{\lambda}^{(n)} = \mathbf{C}_{\mathrm{s}}\boldsymbol{\eta}^{(n)} = \mathbf{C}_{\mathrm{s}}\sum_{l\in\mathcal{C}}\upsilon_l^{(n)}\boldsymbol{\Theta}_l^{(n)}\boldsymbol{\eta}^{(n-1)}. \tag{17}$$

Using eq. (3), it follows that:

$$\begin{aligned}\boldsymbol{\lambda}^{(n)} = &\left(\mathbf{C}_{\mathrm{s}}\sum_{l\in\mathcal{C}}\upsilon_l^{(n)}\boldsymbol{\Theta}_l^{(n)}\mathbf{C}_{\mathrm{s}}^{\mathrm{T}}\left(\mathbf{C}_{\mathrm{s}}\mathbf{C}_{\mathrm{s}}^{\mathrm{T}}\right)^{-1}\right)\boldsymbol{\lambda}^{(n-1)} \\ &+ \mathbf{C}_{\mathrm{s}}\sum_{l\in\mathcal{C}}\upsilon_l^{(n)}\boldsymbol{\Theta}_l^{(n)}\mathbf{n}_C^{(n-1)}.\end{aligned} \tag{18}$$

It can be seen that the mapping from $\boldsymbol{\lambda}^{(n-1)}$ to $\boldsymbol{\lambda}^{(n)}$ is complicated. Specifically, unlike the mapping between two consecutive SCP vectors, which can be simply written as $\boldsymbol{\eta}^{(n)} = \boldsymbol{\Theta}^{(n)}\boldsymbol{\eta}^{(n-1)}$, the mapping between consecutive CCP vectors cannot be written in a linear form due to the second item in eq. (18), i.e., $\mathbf{C}_{\mathrm{s}}\sum_{l\in\mathcal{C}}\upsilon_l^{(n)}\boldsymbol{\Theta}_l^{(n)}\mathbf{n}_C^{(n-1)}$. Moreover, despite that eq. (18) seems to have an affine form, the mapping from $\boldsymbol{\lambda}^{(n-1)}$ to $\boldsymbol{\lambda}^{(n)}$ is not affine either. This is implicitly conveyed through the variable $\mathbf{n}_C^{(n-1)}$ since the value of $\mathbf{n}_C^{(n-1)}$ depends on $\boldsymbol{\lambda}^{(n-1)}$ and the dependence is nonlinear as explained after eq. (3) in Section II.

[1]Here it is assumed that a sufficient number of requests have occurred, i.e., $n\to\infty$.

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications

B. Instantaneous STF - The General Case

Under time-varying content popularity, the state transition probability matrix is $\mathbf{\Theta}^{(n)}$ when the SCP is $\boldsymbol{\eta}^{(n-1)}$. Therefore, the STF at the instant of the nth request and the point $\boldsymbol{\eta}^{(n-1)}$ is given by:

$$\boldsymbol{u}^{(n)}(\boldsymbol{\eta}^{(n-1)}) = \mathbf{\Theta}^{(n)}\boldsymbol{\eta}^{(n-1)} - \boldsymbol{\eta}^{(n-1)}. \tag{19}$$

The superscript (n) in $\boldsymbol{u}^{(n)}(\cdot)$ reflects the fact that the STF is no longer static but time-varying as a result of the time-varying content popularity. The direction and strength of the instantaneous STF depend on both $\boldsymbol{\eta}$, i.e., the location in the state transition domain, and n, i.e., the request instant. The value of the instantaneous STF $\boldsymbol{u}^{(n)}(\boldsymbol{\eta}^{(n-1)})$ represents the change in the SCP after the nth round of request and replacement. The effect of a replacement scheme on the dynamic SCP over a sequence of requests can be decomposed into the summation over the instantaneous STFs:

$$\begin{aligned}\boldsymbol{\eta}^{(n+N-1)} - \boldsymbol{\eta}^{(n-1)} &= \sum_{t=0}^{N-1}\left(\boldsymbol{\eta}^{(n+t)} - \boldsymbol{\eta}^{(n+t-1)}\right)\\ &= \sum_{t=0}^{N-1}\boldsymbol{u}^{(n+t)}(\boldsymbol{\eta}^{(n+t-1)}),\end{aligned} \tag{20}$$

for any $n \geq 1$ and $N \geq 1$.

Similarly, other metrics can also be studied through instantaneous STFs, e.g., the average cache hit probability.

Lemma 2: Using instantaneous STFs from the first till the nth request, the average cache hit probability over the n requests can be given by:

$$\gamma_{\text{avg}} = \frac{1}{n}\sum_{t=2}^{n}\left(\boldsymbol{v}^{(t)}\right)^{\mathrm{T}}\mathbf{C}_{\text{s}}\left(\sum_{t'=0}^{t-2}\boldsymbol{u}^{(t'+1)}\right) + \boldsymbol{v}_{\text{avg}}^{\mathrm{T}}\mathbf{C}_{\text{s}}\boldsymbol{\eta}^{(0)}, \tag{21}$$

in which $\boldsymbol{u}^{(t'+1)}$ is the abbreviation for $\boldsymbol{u}^{(t'+1)}(\boldsymbol{\eta}^{(t')})$, and

$$\boldsymbol{v}_{\text{avg}} = \frac{1}{n}\sum_{t=1}^{n}\boldsymbol{v}^{(t)} \tag{22}$$

is the average content popularity over the n requests.

Proof: See Section B in Appendix.

Lemma 2 shows that the average cache hit probability over an arbitrary number of requests, starting from any initial SCP $\boldsymbol{\eta}^{(\mathbf{0})}$, can be obtained from instantaneous STFs, instantaneous content request probabilities, and the initial point $\boldsymbol{\eta}^{(\mathbf{0})}$. The inner summation over t' in eq. (21) represents the effect of historical requests and replacements on the instantaneous cache hit probability at the tth request. The decomposition in eq. (20) and the result in eq. (21) demonstrate the importance in analyzing the instantaneous STF under different replacement schemes. If the instantaneous content request probabilities $\boldsymbol{v}^{(t)}, t \in \{1, \ldots, n\}$ can be obtained, the instantaneous STF of a replacement scheme at any point in the state transition region can be calculated using eqs. (4), (5), and (19). For evaluating and comparing different cache replacement schemes, we can substitute the specific STF of the replacement schemes for $\boldsymbol{u}^{(1)}, \ldots, \boldsymbol{u}^{(t-1)}$ in eq. (21).

The instantaneous STF can be decomposed. Define the lth component of $\mathbf{u}^{(n)}(\boldsymbol{\eta}^{(n-1)})$ as:

$$\mathbf{u}_l^{(n)} = \mathbf{\Theta}_l^{(n)}\boldsymbol{\eta}^{(n-1)} - \boldsymbol{\eta}^{(n-1)}. \tag{23}$$

It can be seen that:

$$\begin{aligned}\mathbf{u}^{(n)}(\boldsymbol{\eta}^{(n-1)}) =& \mathbf{\Theta}^{(n)}\boldsymbol{\eta}^{(n-1)} - \boldsymbol{\eta}^{(n-1)}\\ =& \sum_{l\in\mathcal{C}} v_l^{(n)}\left(\mathbf{\Theta}_l\boldsymbol{\eta}^{(n-1)} - \boldsymbol{\eta}^{(n-1)}\right)\\ =& \sum_{l\in\mathcal{C}} v_l^{(n)}\mathbf{u}_l^{(n)}.\end{aligned} \tag{24}$$

C. The Case of RR, LP, TLP, and LRU

When a specific replacement scheme is considered, $\mathbf{u}_l^{(n)}$ can be found based on its conditional state transition probability matrix $\mathbf{\Theta}_l^{(n)}$ using (23).

For the case of RR, the mth element of $\mathbf{u}_l^{(n)}$ is given by:

$$u_{m,l,\text{RR}} = \begin{cases}\phi \sum\limits_{\{k|m\in\mathcal{H}_{k,l}\}} \eta_k, & \text{if } l \in \mathcal{C}_m,\\ -L\phi\eta_m, & \text{otherwise.}\end{cases} \tag{25}$$

The mth element of $\mathbf{u}_l^{(n)}$ for LP is given by:

$$u_{m,l,\text{LP}}^{(n)} = \begin{cases}\sum\limits_{k\in\mathcal{G}_{m,l}^{(n)}} \eta_k^{(n-1)}\phi_{l,e(k,m),k}^{(n)}, & \text{if } l \in \mathcal{C}_m,\\ -\eta_m^{(n-1)}, & \text{if } l \notin \mathcal{C}_m \text{ and } \min\limits_{q\in\mathcal{C}_m}\{\tilde{v}_q^{(n+1)}\} < \tilde{v}_l^{(n+1)},\\ 0, & \text{otherwise,}\end{cases} \tag{26}$$

where

$$\mathcal{G}_{m,l}^{(n)} = \{k | m \in \mathcal{H}_{k,l}, \tilde{v}_{e(k,m)}^{(n+1)} < \tilde{v}_l^{(n+1)}\}, \tag{27}$$

representing the set of states which include state m in their content-l neighbors and cache a less popular content compared to state m according to the predicted popularity for the $(n+1)$th request.

Similarly, the mth element of $\mathbf{u}_l^{(n)}$ for TLP is given by:

$$u_{m,l,\text{TLP}}^{(n)} = \begin{cases}\sum\limits_{k\in\hat{\mathcal{G}}_{m,l}^{(n)}} \phi_{e(m,k),q^\dagger(k),k}^{(n)}\eta_k^{(n-1)}, & \text{if } l \in \mathcal{C}_m,\\ -\phi_{e(m,k),q^\dagger(k),k}^{(n)}\eta_m^{(n-1)}, & \\ \qquad\qquad \text{if } l \notin \mathcal{C}_m \text{ and } \min\limits_{q\in\mathcal{C}_m}\{\tilde{v}_q^{(n+1)}\} < \tilde{v}_l^{(n+1)}, & \\ 0, \qquad\qquad \text{otherwise,} & \end{cases} \tag{28}$$

where

$$\hat{\mathcal{G}}_{m,l}^{(n)} = \{k | m \in \mathcal{H}_{k,l}, \tilde{v}_{e(k,m)}^{(n+1)} = \min_{q\in\mathcal{C}_k}\{\tilde{v}_q^{(n+1)}\} < \tilde{v}_l^{(n+1)}\}, \tag{29}$$

representing the set of states which include state m in their content-l neighbors and cache a content less popular than any content cached by state m according to the predicted popularity for the $(n+1)$th request.

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications

For the case of LRU, the mth element of $\mathbf{u}_l^{(n)}$ is given by:

$$u_{m,l,\mathrm{LRU}}^{(n)} = \begin{cases} \sum\limits_{k\in\mathcal{G}_{m,l}} \rho^{(n)}(e(k,m)|k)\eta_k^{(n-1)}, & \text{if } l\in\mathcal{C}_m \\ -\eta_m^{(n-1)}, & \text{otherwise} \end{cases} \tag{30}$$

where

$$\mathcal{G}_{m,l} = \{k | m \in \mathcal{H}_{k,l}\}. \tag{31}$$

In the next section, we study the instantaneous STF of the considered replacement schemes and its impact on their instantaneous cache hit probability.

V. Impact of STF on Instantaneous Cache Hit Probability

When the content popularity varies over time, a replacement scheme may not lead to any steady state. As a result, the analysis of steady states and rate of convergence does not apply. Instead, the impact of a replacement scheme on the instantaneous cache hit probability at the next request is investigated.

A. The General Case

A replacement after the nth request affects the cache hit probability at the $(n+1)$th request. Consider the time instant right after the nth request and replacement so that $\boldsymbol{u}^{(n)}(\cdot)$ is the current STF and the $(n+1)$th request is the next request in future. The effect of a replacement scheme can be conveyed through the difference between the cache hit probability at the $(n+1)$th request with and without a replacement (based on the chosen scheme) after the nth request. This difference is given by:

$$\begin{aligned} d_\gamma^{(n+1)} &= \left(\boldsymbol{v}^{(n+1)}\right)^{\mathrm{T}} \mathbf{C}_{\mathrm{s}} \left(\boldsymbol{\eta}^{(n)} - \boldsymbol{\eta}^{(n-1)}\right) \\ &= \left(\boldsymbol{v}^{(n+1)}\right)^{\mathrm{T}} \mathbf{C}_{\mathrm{s}} \boldsymbol{u}^{(n)}(\boldsymbol{\eta}^{(n-1)}). \end{aligned} \tag{32}$$

The above result shows that, the cache hit ratio at the $(n+1)$th request depends on the content popularity at the $(n+1)$th request, i.e., $\boldsymbol{v}^{(n+1)}$, the STF at the nth request, i.e., $\boldsymbol{u}^{(n)}(\cdot)$, and the SCP at the $(n-1)$th request, i.e., $\boldsymbol{\eta}^{(n-1)}$. Among these three factors, $\boldsymbol{\eta}^{(n-1)}$ reflects the accumulative effect of the previous $n-1$ rounds of request and replacement, $\boldsymbol{u}^{(n)}(\cdot)$ represents the current STF, and $\boldsymbol{v}^{(n+1)}$ represents the content popularity at the next request in future. The result in eq. (32) shows the complication due to time-varying content popularity: $\boldsymbol{v}^{(n+1)}$ and $\boldsymbol{u}^{(n)}(\cdot)$ in eq. (32) would reduce to $\boldsymbol{v}$ and $\boldsymbol{u}(\cdot)$, respectively, if the content popularity becomes time-invariant.

Some general observations can be made:

1) Define $\boldsymbol{z}^{(n+1)} = \mathbf{C}_{\mathrm{s}}^{\mathrm{T}}\boldsymbol{v}^{(n+1)}$. Then $\boldsymbol{z}^{(n+1)}$ is the state cache hit probability vector at the $(n+1)$th request. Depending on $\boldsymbol{\eta}^{(n-1)}$, $\boldsymbol{v}^{(n)}$, and $\boldsymbol{\Theta}^{(n)}$, $\boldsymbol{\eta}^{(n+1)}$ may fall at any point in the areas S_1 in Fig. 1. The replacement after the nth request improves the instantaneous cache hit probability at the $(n+1)$th request if the replacement drives the SCP into the area S_2 shown in Fig. 1.

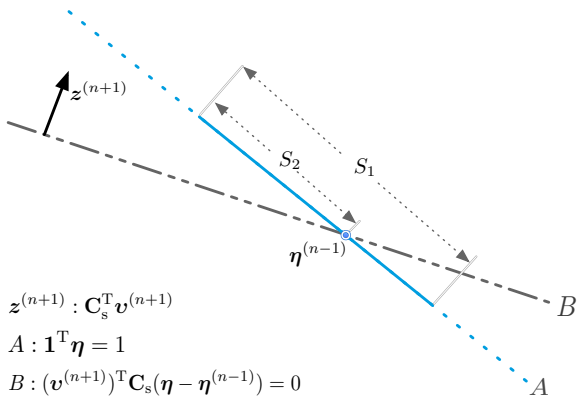


Fig. 1: Illustration of the relation between instantaneous cache hit probability, $\boldsymbol{\eta}^{(n)}$, and $\boldsymbol{v}^{(n+1)}$. Area S_1 is the area that $\boldsymbol{\eta}^{(n+1)}$ may fall in, i.e., the intersection of hyperplane A and the subspace $\boldsymbol{\eta}^{(n+1)} \succeq 0$. If $\boldsymbol{\eta}^{(n+1)}$ falls in area S_2, then $(\boldsymbol{z}^{(n+1)})^T\boldsymbol{\eta}^{(n+1)} \geq (\boldsymbol{z}^{(n+1)})^T\boldsymbol{\eta}^{(n)}$.

2) $d_\gamma^{(n+1)}$ is small, regardless of $\boldsymbol{v}^{(n+1)}$, when $\boldsymbol{\eta}^{(n-1)}$ is close to the steady state corresponding to $\boldsymbol{v}^{(n)}$ (i.e., the steady state if the content popularity is constant and remains equal to $\boldsymbol{v}^{(n)}$).
3) In the trivial case when $\boldsymbol{v}^{(n+1)}$ approaches $1/N_{\mathrm{c}} \cdot \mathbf{1}$, where N_{c} is the number of contents, the hyperplane $(\boldsymbol{v}^{(n+1)})^T\mathbf{C}_{\mathrm{s}}(\boldsymbol{\eta} - \boldsymbol{\eta}^{(n)}) = 0$ coincides with the hyperplane $\mathbf{1}^{\mathrm{T}}\boldsymbol{\eta} = 1$. In such case, $d_\gamma^{(n+1)}$ becomes zero for any replacement scheme.

The effect of a replacement scheme on $d_\gamma^{(n+1)}$ can be conveyed through the set of content-specific instantaneous STF $\{\mathbf{u}_l^{(n)}\}$ using eq. (24).

Theorem 1: The $d_\gamma^{(n+1)}$ in eq.(32) can be equivalently rewritten as:

$$d_\gamma^{(n+1)} = \sum_{l\in\mathcal{C}} (v_l^{(n)} - \bar{v}_l) c_l^{(n+1)}, \tag{33}$$

where

$$c_l^{(n+1)} = \left(\boldsymbol{v}^{(n+1)}\right)^{\mathrm{T}} \mathbf{C}_{\mathrm{s}} \mathbf{u}_l^{(n)}, \tag{34}$$

and $\{\bar{v}_l\}_{l\in\mathcal{C}}$ represents the content popularity under which $\boldsymbol{\eta}^{(n-1)}$ would be the steady state.

Proof: See Section C in Appendix.

Based on eq. (33) and eq. (34), the factors that determine $d_\gamma^{(n+1)}$ are: $\{v_l^{(n+1)}\}$, $\{v_l^{(n)}\}$, $\{\bar{v}_l\}$, and $\mathbf{u}_l^{(n)}$. The factor $\{\bar{v}_l\}$ depends on the historical content requests till the $(n-1)$th request, $\mathbf{u}_l^{(n)}$ depends on $\boldsymbol{\eta}^{(n)}$, and both $\{\bar{v}_l\}$ and $\mathbf{u}_l^{(n)}$ depend on the replacement scheme. The term $v_l^{(n)} - \bar{v}_l$ reflects the deviation in the request probability for content l from its 'steady' request probability, which manifests the influence of historical requests. The term $c_l^{(n+1)}$ represents the change in the cache hit probability at the $(n+1)$th request, using the corresponding replacement scheme, when the current SCP is $\boldsymbol{\eta}^{(n-1)}$ and content l is requested at the nth request.

Using Theorem 1, a more detailed investigation could be conducted for a specific content popularity model (i.e., shot

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications

noise model [15]). Nevertheless, the study on specific content popularity models is not the focus of this work. Section VI, however, will cover numerical results on the performance of replacement schemes under specific content popularity models.

B. Upper and Lower Bounds of $d_\gamma^{(n+1)}$

The term $\mathbf{C}_\mathrm{s}\boldsymbol{u}^{(n)}(\boldsymbol{\eta}^{(n-1)})$ in $d_\gamma^{(n+1)}$ represents the change in the content caching probabilities after the nth request under the chosen replacement scheme. Sort the contents based on their popularity at the instant of the nth request so that $\upsilon_1^{(n)} \geq \upsilon_2^{(n)} \geq \cdots \geq \upsilon_{N_\mathrm{c}}^{(n)}$. The upper-bound and lower-bound of $d_\gamma^{(n+1)}$ can be found using the following result.

Theorem 2: The upper-bound and lower-bound of $d_\gamma^{(n+1)}$, denoted as $\hat{d}_\gamma^{(n+1)}$ and $\check{d}_\gamma^{(n+1)}$, respectively, for RR, LP, TLP, and LRU are given by [2]:

$$\hat{d}_\gamma^{(n+1)} = \begin{cases} L\phi \max\limits_l\{\upsilon_l^{(n)}\}, & \text{RR} \\ \alpha \max\limits_l\{\upsilon_l^{(n)}\}, & \text{LP} \\ \max\limits_l\{\upsilon_l^{(n)}\}, & \text{TLP-A or LRU} \\ \max\limits_l\{\upsilon_l^{(n)}\} \max\limits_l\{\tilde{\upsilon}_l^{(n+1)}\}, & \text{TLP-P} \end{cases} \tag{35}$$

and

$$\check{d}_\gamma^{(n+1)} = \begin{cases} -\phi, & \text{RR} \\ -\alpha, & \text{LP} \\ -1, & \text{TLP-A or LRU} \\ -\sum\limits_{l=1}^{N_\mathrm{c}} \upsilon_l^{(n)}\tilde{\upsilon}_l^{(n+1)}, & \text{TLP-P.} \end{cases} \tag{36}$$

Proof: See Section D in Appendix.

C. Observations

The following observations can be made from the preceding analysis of the relation between the instantaneous STF and the difference in cache hit probability. [3]

- From eq. (25), eq. (33), and eq. (34), it can be seen that the parameter ϕ is only a scaling factor in $d_\gamma^{(n+1)}$ in the case of RR. Specifically, whether $d_\gamma^{(n+1)}$ is negative or not is jointly decided by $\boldsymbol{\upsilon}^{(n+1)}$, $\boldsymbol{\upsilon}^{(n)}$, and $\boldsymbol{\eta}^{(n-1)}$. The parameter ϕ can scale $d_\gamma^{(n+1)}$ but does not have any impact on its sign. This explains the result in [25] that ϕ impacts on the convergence speed but not the steady state under constant content popularity.
- Four cases of instantaneous STF $\mathbf{u}^{(n)}(\boldsymbol{\eta}^{(n-1)})$ and $\boldsymbol{z}^{(n+1)}$ are illustrated in Fig. 2a and Fig. 2b. In each single replacement, both LP and TLP drive the SCP $\boldsymbol{\eta}$ towards a direction that increases $(\boldsymbol{z}^{(n+1)})^\mathrm{T}\boldsymbol{\eta}$, i.e., $(\boldsymbol{z}^{(n+1)})^\mathrm{T}\boldsymbol{\eta}^{(n+1)} \geq (\boldsymbol{z}^{(n+1)})^\mathrm{T}\boldsymbol{\eta}^{(n)}$, where $\boldsymbol{z}^{(n+1)} = \mathbf{C}_\mathrm{s}^\mathrm{T}\boldsymbol{\upsilon}^{(n+1)}$. Therefore, only case 2 in Fig. 2a and case 3 in Fig. 2b are possible for LP and TLP while all four cases can occur for RR and LRU. Moreover, TLP drives $\boldsymbol{\eta}$ towards the direction that increases $(\boldsymbol{z}^{(n+1)})^\mathrm{T}\boldsymbol{\eta}$ the fastest, which is a resemblance to the steepest gradient in optimization. This explains the result in the first part that TLP converges faster than LP under constant content popularity.
- Under time-varying content popularity, LP and TLP may not effectively trace the varying content popularity depending on the pattern of variation. Specifically, if $\boldsymbol{\upsilon}^{(n)}$ varies so that $\boldsymbol{z}^{(n)}$ changes along a straight path over time, as shown in Fig. 2c, then LP and TLP can still trace the content popularity well, and TLP should outperform LP. An example of such scenario is when popularity concentrates so that the most popular contents become even more popular over time.
- If $\boldsymbol{\upsilon}^{(n)}$ varies so that $\boldsymbol{z}^{(n)}$ changes fast and randomly in an area, as shown in Fig. 2d, then LP and TLP may not trace the content popularity well, and TLP can perform worse than LP. An example of such scenario is when content popularity varies drastically over time so that the most popular set of contents rapidly changes.

VI. Numerical Examples

A. Instantaneous STF under Time-varying Content Popularity

Fig. 3 demonstrates the instantaneous STF under time-varying content popularity and further illustrates Fig. 1 using RR and LP as examples. Similar to the first part of this two-part paper, we use 3-D STFs for illustrations.

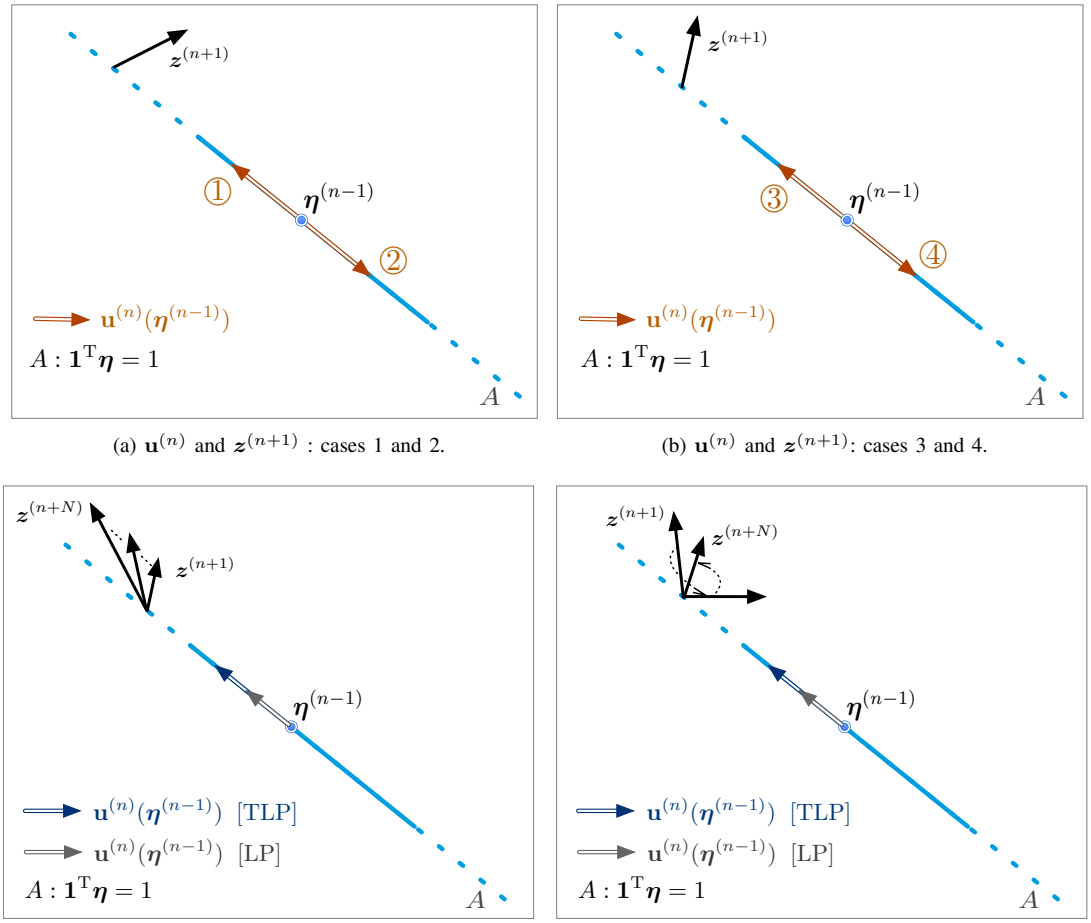
Fig. 3a shows the case under RR. The content popularity at the nth and $(n+1)$th requests are $\boldsymbol{\upsilon}^{(n)} = [0.46, 0.30, 0.24]^\mathrm{T}$ and $\boldsymbol{\upsilon}^{(n+1)} = [0.4, 0.35, 0.25]^\mathrm{T}$, respectively. The solid circle with red filling shows where the steady state would be if the content popularity were fixed and equal to $\boldsymbol{\upsilon}^{(n)}$. The hollow circle shows where the stationary state would be if the content popularity were fixed and equal to $\boldsymbol{\upsilon}^{(n+1)}$. The black triangular area with solid edges represents the state transition domain. The black arrows demonstrate the direction and strength of the STF at the instant of the nth request and the corresponding locations in the state transition domain. The colored straight lines in the x-y plane show the contour of the cache hit probability in the state transition domain. The solid straight line from the origin $(0, 0, 0)$ to the diamond marker in the STF are specified by the vector $\mathbf{C}_\mathrm{s}\boldsymbol{\upsilon}^{(n+1)}$. Denote the SCP vector $\boldsymbol{\eta}$ at the diamond marker as $\bar{\boldsymbol{\eta}}^{(n)}$. The dashed triangle in blue represents the intersection of the plane $(\boldsymbol{\upsilon}^{(n+1)})^\mathrm{T}\mathbf{C}_\mathrm{s}(\boldsymbol{\eta} - \bar{\boldsymbol{\eta}}^{(n)}) = 0$ with the 3 planes $\eta_1 = 0$, $\eta_2 = 0$, and $\eta_3 = 0$. The dotted line represents the intersection of the plane $(\boldsymbol{\upsilon}^{(n+1)})^\mathrm{T}\mathbf{C}_\mathrm{s}(\boldsymbol{\eta} - \bar{\boldsymbol{\eta}}^{(n)}) = 0$ with the state transition domain.

From Fig. 3a, the effect of the nth replacement, given the replacement scheme of RR and the above change of content popularity from $\boldsymbol{\upsilon}^{(n)}$ to $\boldsymbol{\upsilon}^{(n+1)}$, can be observed. Specifically, given any SCP, i.e, a point in the state transition domain, if the arrow representing the instantaneous STF at that point can be scaled such that it crosses the dotted line from below to above, the nth replacement yields a smaller cache hit probability at the $(n+1)$th request compared with no replacement. By

[2] For the lower-bound of $d_\gamma^{(n+1)}$ in the case of TLP-P, it is assumed that the L least popular contents at the nth request remain least popular at the $(n+1)$th request.

[3] Accurate prediction of content popularity is assumed for the case of LP and TLP.

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications



(a) $\mathbf{u}^{(n)}$ and $\boldsymbol{z}^{(n+1)}$: cases 1 and 2.

(b) $\mathbf{u}^{(n)}$ and $\boldsymbol{z}^{(n+1)}$: cases 3 and 4.

(c) $\boldsymbol{z}^{(n)}$ changes along a straight path.

(d) $\boldsymbol{z}^{(n)}$ changes randomly.

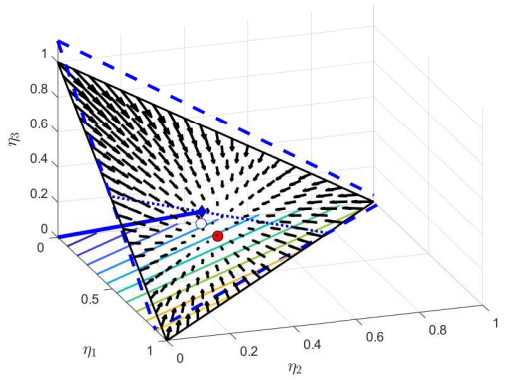
Fig. 2: Illustration of the relation between the replacement schemes, the instantaneous STF $\mathbf{u}^{(n)}(\boldsymbol{\eta}^{(n-1)})$, and the state cache hit probability $\boldsymbol{z}^{(n+1)}$.

contrast, if the arrow can be scaled such that it crosses the dotted line from above to below, the nth replacement yields a larger cache hit probability at the $(n+1)$th request. If the arrow is in parallel with the dotted line, the nth replacement has no impact on the cache hit probability at the $(n+1)$th request.

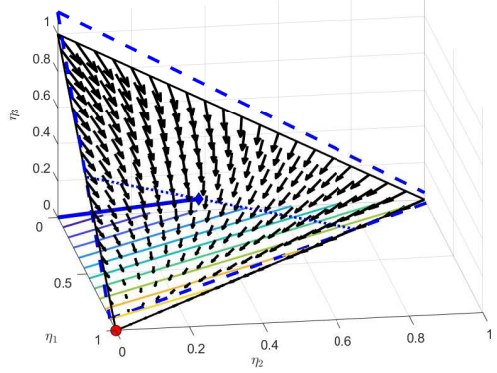
Fig. 3b shows the first of two examples with LP. The content popularity $\boldsymbol{\upsilon}^{(n)}$ and $\boldsymbol{\upsilon}^{(n+1)}$ are the same as in Fig. 3a. In this example, the change in the content popularity is not significant so that the state which caches the most popular contents does not change. As a result, the stationary state if the content popularity is fixed and equal to $\boldsymbol{\upsilon}^{(n)}$ and that if the content popularity is fixed and equal to $\boldsymbol{\upsilon}^{(n+1)}$ are identical and shown by a solid circle in the figure. The dashed triangle, solid straight line, and dotted line illustrate the same objects or variables as in Fig. 3a, respectively. The effect of the nth replacement on the cache hit probability at the $(n+1)$th request at any SCP point in the state transition domain can be observed from Fig. 3b following the same method described in the preceding paragraph. In this example, the arrow at any point (except the stationary point) can be scaled such that it crosses the dotted line with the arrow head below the line. As a result, a replacement after request n based on LP always increases the cache hit probability at the $(n+1)$th request (except at $\boldsymbol{\eta} = [1, 0, 0]$). This example corresponds to the scenario of varying content popularity which drives $\boldsymbol{z}^{(n)}$ along a somewhat straight path, as shown in Fig. 2c.

Fig. 3c shows the second example with LP. The content popularity $\boldsymbol{\upsilon}^{(n)}$ is the same as in Fig. 3a and Fig. 3b, while $\boldsymbol{\upsilon}^{(n+1)} = [0.4, 0.25, 0.35]^{\mathrm{T}}$. The solid and the hollow circles show the stationary states in the cases when the content popularity is fixed and equal to $\boldsymbol{\upsilon}^{(n)}$ and $\boldsymbol{\upsilon}^{(n+1)}$, respectively. At any SCP point, if the arrow can be scaled such that it crosses the dotted line from right to left, the nth replacement yields a smaller cache hit probability at the $(n+1)$th request compared with no replacement. By contrast, if the arrow can be scaled such that it crosses the dotted line from left to right, the nth replacement yields a larger cache hit probability at the $(n+1)$th request. In this example, a replacement after request n based on LP may either increase or decrease the cache hit probability at the $(n+1)$th request. This example corresponds to the scenario of varying content popularity which leads to a randomly changing $\boldsymbol{z}^{(n)}$, as shown in Fig. 2d.

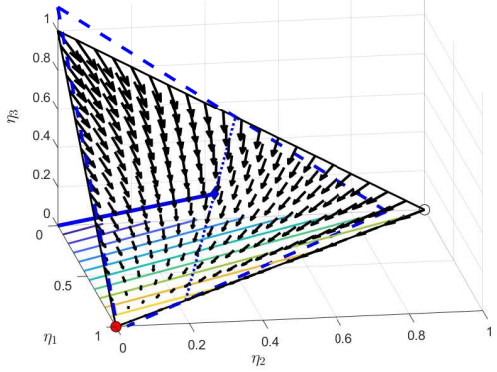
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2019.2940667, IEEE Transactions on Wireless Communications



(a) RR.

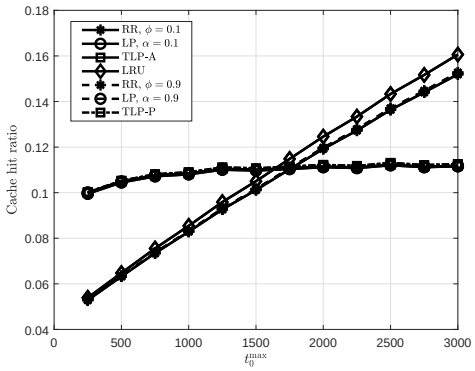


(b) LP, example 1.

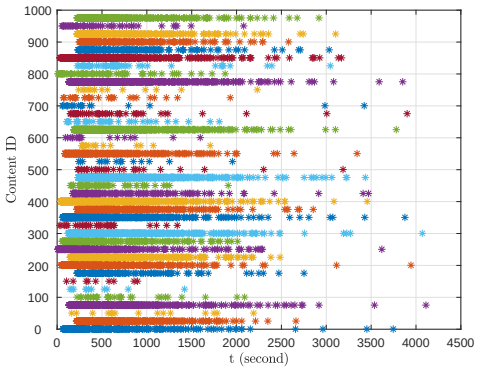


(c) LP, exapmle 2.

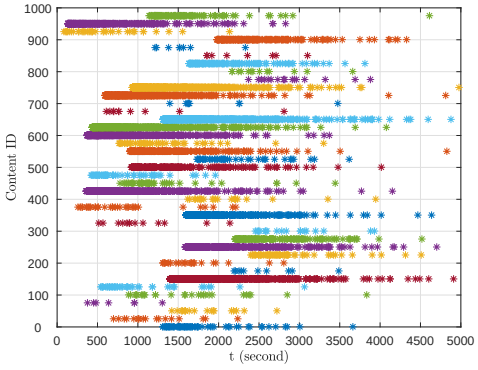
Fig. 3: Instantaneous STF and its impact on the instantaneous cache hit probability at the next request.



(a) Cache hit ratio versus $t_0^{\max}$.



(b) Request instants for 40 out of 1000 contents when $t_0^{\max} = 250$.



(c) Request instants for 40 out of 1000 contents in one round when $t_0^{\max} = 2500$.

Fig. 4: Cache hit ratio under shot noise model.

B. Cache Hit Ratio under Time-varying Content Popularity

In the second set of examples, the cache hit ratio of the considered cache replacement schemes under time-varying content popularity is demonstrated.

First, the cache hit ratio is demonstrated when the time-varying content popularity is generated using the shot noise model [15]. Specifically, the request for content l follows a time-inhomogeneous Poisson process with the instantaneous rate at time t given by:

$$y_l(t) = \begin{cases} A_l b_l \exp^{-b_l(t-t_{l,0})}, & \text{if } t \geq t_{l,0} \\ 0, & \text{otherwise} \end{cases} \quad (37)$$

Accordingly, requests for content l start occurring from $t_{l,0}$. The parameter A_l limits the maximum request rate of content l. For content l, an allocation of A_l over time is given by an exponential distribution with rate parameter b_l. It follows that contents have different life-span and entrance time. The entrance time $t_{l,0}$ is uniformly generated in $[0, t_0^{\max}]$, and A_l is uniformly generated in $[A_l^{\min}, A_l^{\max}]$. For RR, we test two cases, $\phi = 0.9$ and $\phi = 0.1$. A larger ϕ results in more frequent

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications

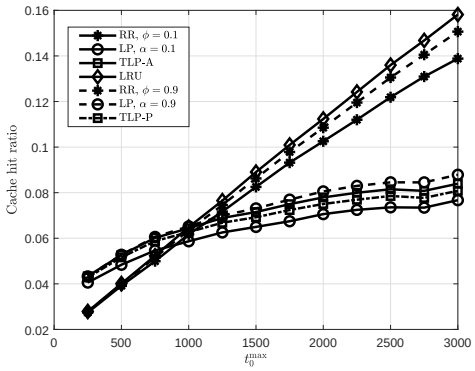
content replacements and higher sensitivity to the changes in the content popularity. Similarly, for LP, we test two cases, i.e., $\alpha = 0.9$ and $\alpha = 0.1$.

In the first example with shot noise model, the number of contents N_c is set to 1000 and the cache size L is set to 15. A duration with 5000 seconds from $t = 0$ to $t = 5000$ is considered. The parameters $A_l^{\min}$ and $A_l^{\max}$ are set to 10 and 1000, respectively. Fig. 4a shows the resulting cache hit ratio of the considered replacement schemes versus $t_0^{\max}$. Each data point in Fig. 4a is averaged over 200 rounds of simulations for the considered 5000 seconds duration. For LP and TLP, accurate prediction of content popularity is assumed. It can be seen from the Fig. 4a that LP and TLP have a significant advantage over RR and LRU when $t_0^{\max}$ is small (i.e., $t_0^{\max} \leq 1000$). However, RR and LRU are much better than LP and TLP when $t_0^{\max}$ becomes large.

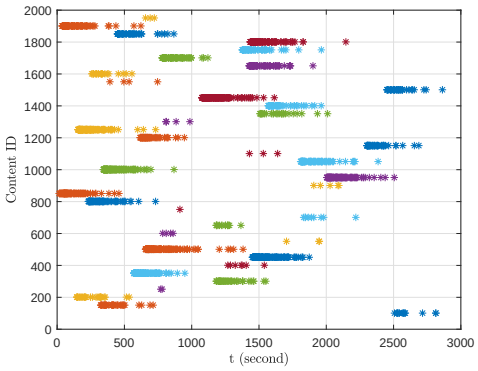
The content request time instants for 40 out of the 1000 contents [4] in the case when $t_0^{\max} = 250$ and $t_0^{\max} = 2500$ are plotted in Figs. 4b and 4c, respectively. Colors are used to distinguish the requests for different contents. Each asterisk in Figs. 4b and 4c represents a request, with its x and y coordinates specifying the corresponding request time instant and the content ID, respectively. It can be seen from Figs. 4b and 4c that, when $t_0^{\max}$ becomes large, the set of available contents can vary significantly over time. This has two effects on the cache hit ratio. On one hand, the cache hit ratio should increase as the number of simultaneous available contents can be smaller when $t_0^{\max}$ is large. On the other hand, due to the property of the instantaneous rate given by eq. (37), the maximum instantaneous request rate of any content occurs when the content just becomes available. If follows that the varying set of available contents when $t_0^{\max}$ is large can lead to frequent and abrupt change of content popularity over time, as illustrated in Fig. 2d and Fig. 3c. Since LP and TLP exploit the content popularity information in a greedy manner (i.e., maximizing the cache hit ratio based on the current content popularity information), the second effect can hinder the cache hit ratio, and the combined impact of the above two effects yields an almost steady cache hit ratio of LP and TLP in Fig. 4a. By contrast, the cache hit ratio of RR and LRU increases with $t_0^{\max}$ as the result of the first effect while the second effect has no significant impact as RR and LRU do not rely on the instantaneous content popularity information.

In the second example with shot noise model, N_c is increased from 1000 to 2000, and $A_l^{\max}$ and $A_l^{\min}$ are decreased from 1000 to 200 and from 10 to 1, respectively. The average content life-span also becomes shorter. Fig. 5a shows the resulting cache hit ratio versus $t_0^{\max}$, while the request time instants for 40 out of the 2000 contents when $t_0^{\max} = 2500$ is plotted in Fig. 5b. Comparing Fig. 5a with Fig. 4a, three observations can be made. First, the cache hit ratio in Fig. 5a becomes lower for all schemes when $t_0^{\max} = 0$, as a result of N_c increasing to 2000. Second, the effect of ϕ and α on the performance of RR and LP, respectively, becomes obvious in Fig. 5a. This is because a larger ϕ or α allows for a faster adaption to new content requests, which is important

[4]Specifically, the contents whose content ID is a multiple of 25 are selected.



(a) Cache hit ratio versus $t_0^{\max}$.



(b) Request instants for 40 out of 2000 contents in one round when $t_0^{\max} = 2500$.

Fig. 5: Cache hit ratio under shot noise model, short life-span.

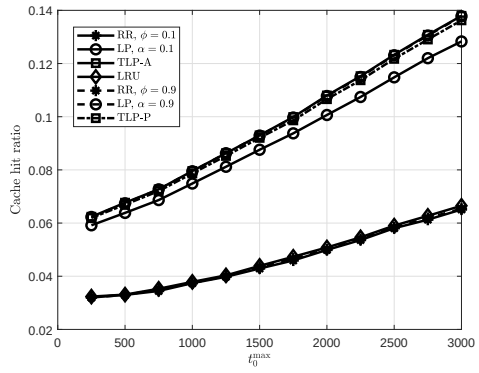
now that the number of requests for each content decreases significantly. Third, RR and LRU begin to outperform LP and TLP from a smaller $t_0^{\max}$ in Fig. 5a compared to that in Fig. 4a, and the performance gap between the two groups becomes larger. This is because the combination of active contents and their popularity varies even more rapidly compared with the case in Fig. 4a, as a result of a larger N_c and shorter content life span. The result in Fig. 5a shows that exploiting the instantaneous content popularity information in a content replacement scheme is not necessarily beneficial for increasing the cache hit ratio even if such information is predicted perfectly. This is because the usefulness of the instantaneous content popularity information depends on how rapidly the content popularity changes. This example corresponds to the case illustrated in Fig. 2d.

Fig. 6 shows the cache hit ratio with a time-varying content popularity model different from eq. (37). Specifically, the request for content l follows a time-inhomogeneous Poisson process with the instantaneous rate at time t given by:

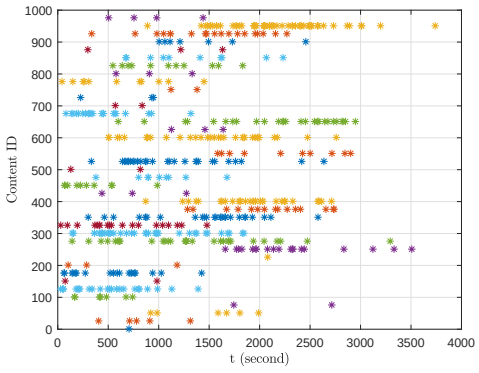
$$y_l(t) = A_l \frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{(t-t_{l,0})}{2\sigma^2}}. \quad (38)$$

The parameter $t_{l,0}$ is no longer the entrance time of instant l in eq. (38). However, $t_{l,0}$ in both eq. (37) and eq. (38) corresponds to the time instant of the peak instantaneous request rate for content l. Similarly, $t_{l,0}$ is uniformly generated in $[0, t_0^{\max}]$, and A_l is uniformly generated in $[A_l^{\min}, A_l^{\max}]$.

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications



(a) Cache hit ratio versus $t_0^{\max}$.



(b) Request instants for 40 out of 1000 contents in one round when $t_0^{\max} = 2500$.

Fig. 6: Cache hit ratio under time-inhomogeneous Poisson process represented by eq. (38).

In this simulation, N_c is set to 1000 and the cache size L is set to 15. A duration with 5000 seconds from $t = 0$ to $t = 5000$ is considered. The parameters $A_l^{\min}$ and $A_l^{\max}$ are set to 1 and 50, respectively. Fig. 6a shows the resulting cache hit ratio of the considered replacement schemes versus $t_0^{\max}$, in which each data point is averaged over 200 rounds of simulations. Accurate prediction of content popularity is again assumed for LP and TLP. The request time instants for 40 out of the 1000 contents when $t_0^{\max} = 2500$ is plotted in Fig. 6b. It can be seen that Fig. 6a shows a very different result when compared with Fig. 4a or Fig. 5a. Specifically, LP and TLP always perform better that RR and LRU in Fig. 6a, and the performance gap between the two groups increases with $t_0^{\max}$. This is because that, unlike the abrupt and frequent variations introduced by eq. (37), the instantaneous rate model in eq.(38) leads to smooth and graduate variations in the content popularity. As a result, the instantaneous content popularity at any instant can be close to the instantaneous content popularity for a number of subsequent requests. Therefore, the greedy maximization of the cache hit ratio based on the current content popularity information used by LP and TLP can benefit the cache hit ratio for both the immediate next request and also subsequent requests. Consequently, the LP and TLP outperform RR and LRU due to the exploration of the instantaneous content popularity information in such case. This example corresponds to the case illustrated in Fig. 2c.

VII. Conclusion

We have extended the study of dynamic caching via STF to the case of time-varying content popularity. In our analysis, we have focused on developing the model and methodology without assuming a specific pattern of change in content popularity. The results have demonstrated the impact of varying popularity on the STF and the performance of replacement schemes in the general case. Further extensions can be conducted by incorporating a specific model of time-varying content popularity. In our simulations, we have adopted different models of varying popularity, and the numerical results have been shown to be consistent with the observations from the analysis.

Through the two parts of this paper, we have provided a novel perspective and developed methods for studying cache replacement in the vector space of SCP using STF. It has been demonstrated that the design of replacement schemes is essentially the design of STF and that the knowledge of content popularity is beneficial only if exploited properly, depending on the pattern of change in the content popularity. As there are many open issues, especially in the case of time-varying content popularity, the results of this paper have been developed in the effort of inspiring the analysis or design of cache replacement schemes for various specific problems and scenarios.

Appendix

A. Proof of Lemma 1

Suppose that the LRU content at the nth request is content $q^\star$, and the most recent request for $q^\star$ is the $(n-w)$th request. It must hold that $w \geq L$, and all requests from $(n-w+1)$th request to the $(n-1)$th request must be for contents $l \in \mathcal{C}_k \backslash \{q^\star\}$. Denote the N_c contents in $l \in \mathcal{C}_k \backslash \{q^\star\}$ as $k(1, \bar{q}^\star), \ldots k(L-1, \bar{q}^\star)$. To allocate the total number of $w-1$ requests (i.e., from the $(n-w+1)$th request to the $(n-1)$th request) to the $L-1$ contents in $l \in \mathcal{C}_k \backslash \{q^\star\}$, there are $P_w = \binom{w-1}{L-1}$ different different allocations, without considering the order of requests, that guarantees at least one request for each content. Denote the number of requests for content $k(i, \bar{q}^\star)$ in the jth combination as $T(k, i, j, \bar{q}^\star)$, where $i \in \{1, \ldots, L-1\}$ and $j \in \{1, \ldots, P_w\}$. It follows that:

$$\sum_{1=1}^{L-1} T(k, i, j, \bar{q}^\star) = w - 1, \forall j. \tag{39}$$

Then, considering the order of request, the number of different ordered allocations are:

$$U_w = \sum_{j=1}^{P_w} \prod_{i=1}^{L-1} \binom{w-1-a_{k,i,j,\bar{q}^\star}}{T(k, i, j, \bar{q}^\star)}. \tag{40}$$

in which

$$a_{k,i,j,\bar{q}^\star} = \begin{cases} 0, & \text{if } i = 1 \\ \sum_{y=1}^{i-1} T(k, y, j, \bar{q}^\star), & \text{if } i \geq 2. \end{cases} \tag{41}$$

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications

Denote the set of request instants for content $k(i,\bar{q}^\star)$ in the uth ordered combination as $\mathcal{T}(k,i,u,\bar{q}^\star)$, where $i \in \{1,\ldots,L-1\}$ and $u \in \{1,\ldots,U_w\}$. It follows that:

$$\bigcup_{i=1}^{L-1} \mathcal{T}(k,i,u,\bar{q}^\star) = \{n-1,\ldots,n-w+1\}, \forall u. \tag{42}$$

Accordingly, the joint probability that the current state is k, content $q^\star = e(k,m) \in \mathcal{C}_k$ is the LRU content at the nth request, and the most recent request for the LRU is the $(n-w)$th request is given by eq. (13). ■

B. Proof of Lemma 2

The average cache hit probability from the 1st till the nth request is given by:

$$\begin{aligned}\gamma_{\text{avg}} &= \frac{1}{n}\sum_{t=1}^{n}\left(\boldsymbol{v}^{(t)}\right)^{\mathrm{T}}\boldsymbol{\lambda}^{(t-1)} \\ &= \frac{1}{n}\sum_{t=1}^{n}\left(\boldsymbol{v}^{(t)}\right)^{\mathrm{T}}\mathbf{C}_{\mathrm{s}}\boldsymbol{\eta}^{(t-1)}.\end{aligned} \tag{43}$$

Using eq. (20) (and setting $n=1$ and $N=t-1$ in eq. (20)), it holds that:

$$\boldsymbol{\eta}^{(t-1)} = \sum_{t'=0}^{t-2}\boldsymbol{u}^{(t'+1)}(\boldsymbol{\eta}^{(t')}) + \boldsymbol{\eta}^{(0)} \tag{44}$$

for any $t \geq 2$. Substituting eq. (20) into eq. (43), it holds that:

$$\begin{aligned}\gamma_{\text{avg}} =& \frac{1}{n}\sum_{t=2}^{n}\left(\boldsymbol{v}^{(t)}\right)^{\mathrm{T}}\mathbf{C}_{\mathrm{s}}\left(\sum_{t'=0}^{t-2}\boldsymbol{u}^{(t'+1)}(\boldsymbol{\eta}^{(t')}) + \boldsymbol{\eta}^{(0)}\right) \\ &+ \frac{1}{n}\left(\boldsymbol{v}^{(1)}\right)^{\mathrm{T}}\mathbf{C}_{\mathrm{s}}\boldsymbol{\eta}^{(0)} \\ =& \frac{1}{n}\sum_{t=2}^{n}\left(\boldsymbol{v}^{(t)}\right)^{\mathrm{T}}\mathbf{C}_{\mathrm{s}}\left(\sum_{t'=0}^{t-2}\boldsymbol{u}^{(t'+1)}(\boldsymbol{\eta}^{(t')})\right) \\ &+ \frac{1}{n}\sum_{t=1}^{n}\left(\boldsymbol{v}^{(t)}\right)^{\mathrm{T}}\mathbf{C}_{\mathrm{s}}\boldsymbol{\eta}^{(0)} \\ =& \frac{1}{n}\sum_{t=2}^{n}\left(\boldsymbol{v}^{(t)}\right)^{\mathrm{T}}\mathbf{C}_{\mathrm{s}}\left(\sum_{t'=0}^{t-2}\boldsymbol{u}^{(t'+1)}(\boldsymbol{\eta}^{(t')})\right) \\ &+ \left(\frac{1}{n}\sum_{t=1}^{n}\boldsymbol{v}^{(t)}\right)^{\mathrm{T}}\mathbf{C}_{\mathrm{s}}\boldsymbol{\eta}^{(0)},\end{aligned} \tag{45}$$

which leads to eq. (21). ■

C. Proof of Theorem 1

As $\bar{v}_l$ is defined such that $\boldsymbol{\eta}^{(n-1)}$ would be the steady state if the content request probabilities were time-invariant and equal to $\{\bar{v}_l\}$. It follows that:

$$\sum_{l\in\mathcal{C}}\bar{v}_l\boldsymbol{\Theta}_l\boldsymbol{\eta}^{(n-1)} = \boldsymbol{\eta}^{(n-1)}. \tag{46}$$

Based on eq. (24) and eq. (46), it holds that:

$$\begin{aligned}\boldsymbol{\eta}^{(n)} - \boldsymbol{\eta}^{(n-1)} &= \sum_{l\in\mathcal{C}}\left(v_l^{(n)} - \bar{v}_l\right)\boldsymbol{\Theta}_l\boldsymbol{\eta}^{(n-1)} \\ &= \sum_{l\in\mathcal{C}}\left(v_l^{(n)} - \bar{v}_l\right)(\boldsymbol{\eta}^{(n-1)} + \mathbf{u}_l^{(n)}) \\ &= \sum_{l\in\mathcal{C}}\left(v_l^{(n)} - \bar{v}_l\right)\mathbf{u}_l^{(n)},\end{aligned} \tag{47}$$

where the last equality uses the fact that $\sum_{l\in\mathcal{C}}\left(v_l^{(n)} - \bar{v}_l\right) = 0$.

Substituting the above equation into eq. (32) gives

$$d_\gamma^{(n+1)} = \left(\boldsymbol{v}^{(n+1)}\right)^{\mathrm{T}}\mathbf{C}_{\mathrm{s}}\sum_{l\in\mathcal{C}}\left(v_l^{(n)} - \bar{v}_l\right)\mathbf{u}_l^{(n)}. \tag{48}$$

Rearranging the above equation using eq. (34) leads to eq. (33). ■

D. Proof of Theorem 2

The proof is based on the equality $d_\gamma^{(n+1)} = \left(\boldsymbol{v}^{(n+1)}\right)^{\mathrm{T}}\mathbf{C}_{\mathrm{s}}\boldsymbol{u}^{(n)}$ in eq. (32). The elements of the $N_{\mathrm{c}}\times 1$ vector $\mathbf{C}_{\mathrm{s}}\boldsymbol{u}^{(n)}$ are the changes in the caching probabilities of the N_{c} contents after the nth request and replacement. It is straightforward to see that the upper and lower bounds of $d_\gamma^{(n+1)}$ are decided by the maximum and minimum elements of $\mathbf{C}_{\mathrm{s}}\boldsymbol{u}^{(n)}$, respectively.

Given that contents are sorted based on their popularity at the nth request, the maximum element of $\mathbf{C}_{\mathrm{s}}\boldsymbol{u}^{(n)}$ for all cases but TLP-P corresponds to the case when content 1 is requested while it is being cached with probability zero. Using eq. (24) and eqs. (25) - (30), it can be seen that the maximum element of $\mathbf{C}_{\mathrm{s}}\boldsymbol{u}^{(n)}$ is $L\phi\max_l\{v_l^{(n)}\}$, $\alpha\max_l\{v_l^{(n)}\}$, $\max_l\{v_l^{(n)}\}$, and $\max_l\{v_l^{(n)}\}$ for RR, LP, TLP-A, and LRU, respectively. For the case of TLP-P, it holds that

$$\hat{d}_\gamma^{(n+1)} \leq \max_l\{v_l^{(n)}\}\cdot\max_l\{\tilde{v}_l^{(n+1)} - \tilde{v}_{N_{\mathrm{c}}}^{(n+1)}\}. \tag{49}$$

For all cases but TLP-P, the minimum of $\mathbf{C}_{\mathrm{s}}\boldsymbol{u}^{(n)}$ corresponds to the following scenario: i). the state with the L least popular contents is being cached with probability 1; and ii). a content not in the cache is requested. The change in the SCP of this state in the described scenario gives the minimum of $\mathbf{C}_{\mathrm{s}}\boldsymbol{u}^{(n)}$.

For RR, the change in the above SCP is given by

$$\check{d}_\gamma^{(n+1)} = -\phi\sum_{l=1}^{N_{\mathrm{c}}-L}v_l^{(n)} \geq -\phi, \tag{50}$$

where the inequality is based on the approximation that the summation of request probabilities of all but the L least popular contents should be close to 1.

For LP, the change is given by

$$\begin{aligned}\check{d}_\gamma^{(n+1)} &= -\alpha\sum_{l=1}^{N_{\mathrm{c}}-L}v_l^{(n)}\frac{\tilde{v}_l^{(n+1)} - \tilde{v}_{N_{\mathrm{c}}}^{(n+1)}}{\sum_{q=N_{\mathrm{c}}-L+1}^{N_{\mathrm{c}}}(\tilde{v}_l^{(n+1)} - \tilde{v}_q^{(n+1)})} \\ &\geq -\alpha\sum_{l=1}^{N_{\mathrm{c}}-L}v_l^{(n)} \geq -\alpha.\end{aligned} \tag{51}$$

For both TLP-A and LRU, the state will change as long as the requested content is not in the cache. Therefore, the aforementioned change is given by

$$\check{d}_{\gamma}^{(n+1)} = -\sum_{l=1}^{N_c-L} \upsilon_l^{(n)} \geq -1. \tag{52}$$

For TLP- P, assuming that the L least popular contents at the nth request remain to be the least popular at the $(n+1)$th request, the change is given by

$$\begin{aligned}\check{d}_{\gamma}^{(n+1)} &= -\sum_{l=1}^{N_c-L} \upsilon_l^{(n)} \left(\tilde{\upsilon}_l^{(n+1)} - \tilde{\upsilon}_{N_c}^{(n+1)}\right) \\ &\geq -\sum_{l=1}^{N_c-L} \upsilon_l^{(n)} \tilde{\upsilon}_l^{(n+1)} \geq -\sum_{l=1}^{N_c} \upsilon_l^{(n)} \tilde{\upsilon}_l^{(n+1)}. \end{aligned} \tag{53}$$

This completes the proof of Theorem 2. ■

REFERENCES

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

[2] S. Zhang, W. Quan, J. Li, W. Shi, P. Yang, and X. Shen, "Air-Ground Integrated Vehicular Network Slicing With Content Pushing and Caching," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2114–2127, Sept. 2018.

[3] J. Gao, L. Zhao, and L. Sun, "Probabilistic Caching as Mixed Strategies in Spatially-Coupled Edge Caching," in *Proc. 29th Biennial Symp. Commun.*, Toronto, Canada, 2018.

[4] S. Müller, O. Atan, M. van der Schaar, and A. Klein, "Context-Aware Proactive Content Caching With Service Differentiation in Wireless Networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 2, pp. 1024–1036, Feb. 2017.

[5] P. Yang, N. Zhang, S. Zhang, L. Yu, J. Zhang, and X. Shen, "Content Popularity Prediction Towards Location-Aware Mobile Edge Caching," *IEEE Trans. Multimedia*, vol. 21, no. 4, pp. 915–929, Apr. 2019.

[6] J. Gao, S. Zhang, L. Zhao, and X. Shen, "The Design of Dynamic Probabilistic Caching with Time-Varying Content Popularity," submitted to *IEEE Trans. Mobile Comput.*, under review.

[7] 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.

[8] G. Paschos, E. Bastug, I. Land, G. Caire, and M. Debbah, "Wireless Caching: Technical Misconceptions and Business Barriers," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 16–22, Aug. 2016.

[9] G. S. Paschos, G. Iosifidis, M. Tao, D. Towsley, and G. Caire, "The Role of Caching in Future Communication Systems and Networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1111–1125, June 2018.

[10] S. M. Azimi, O. Simeone, A. Sengupta, and R. Tandon, "Online Edge Caching and Wireless Delivery in Fog-Aided Networks With Dynamic Content Popularity," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1189–1202, June 2018.

[11] S. Jin and A. Bestavros, "Temporal Locality in Web Request Streams: Sources, Characteristics, and Caching Implications," Technical Report, BUCS-1999-014, Boston, USA, Oct. 1999.

[12] M. Busari and C. Williamson, "On the Sensitivity of Web Proxy Cache Performance to Workload Characteristics," in *Proc. IEEE INFOCOM*, Anchorage, USA, 2001, pp. 1225–1234.

[13] Y. Zhou, L. Chen, C. Yang, and D. M. Chiu, "Video Popularity Dynamics and Its Implication for Replication," *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1273–1285, Aug. 2015.

[14] V. Phalke and B. Gopinath, "An Inter-Reference Gap Model for Temporal Locality in Program Behavior," in *Proc. ACM SIGMETRICS/PERFORMANCE Conf.*, Ottawa, Canada, May 1995, pp. 291–300.

[15] S. Traverso, M. Ahmed, M. Garetto, P. Giaccone, E. Leonardi, and S. Niccolini, "Temporal Locality in Today's Content Caching: Why It Matters and How to Model It," *ACM SIGCOMM Comput. Commun. Rev.* vol. 43, no. 5, pp. 5–12. Nov. 2013.

[16] S. Traverso, M. Ahmed, M. Garetto, P. Giaccone, E. Leonardi, and S. Niccolini, "Unravelling the Impact of Temporal and Geographical Locality in Content Caching Systems," *IEEE Trans. Multimedia*, vol. 17, no. 10, pp. 1839-1854, Oct. 2015.

[17] M. Garetto, E. Leonardi, and V. Martina, "A Unified Approach to the Performance Analysis of Caching Systems," *ACM Trans. Model. Perform. Eval. Comput. Syst.*, vol. 1, no. 3, May 2016.

[18] M. Garetto, E. Leonardi, and S. Traverso, "Efficient Analysis of Caching Strategies under Dynamic Content Popularity," in *Proc. IEEE INFOCOM*, Kowloon, China, Apr./May 2015, pp. 2263–2271.

[19] A. Jaleel, K. B. Theobald, S. C. Steely Jr., and J. Emer, "High Performance Cache Replacement Using Re-reference Interval Prediction (RRIP)," *ACM SIGARCH Comput. Archit. News*, vol. 38, no. 3, pp. 60–71, June 2010.

[20] S. Li, J. Xu, M. van der Schaar, and W. Li, "Popularity-driven content caching," in *Proc. IEEE INFOCOM*, San Francisco, USA, 2016.

[21] N. Zhang, K. Zheng, and M. Tao, "Using Grouped Linear Prediction and Accelerated Reinforcement Learning for Online Content Caching," in *Proc. IEEE ICC Workshops*, Kansas City, USA, May 2018.

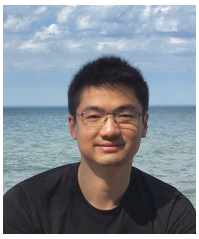
[22] A. Sadeghi, F. Sheikholeslami, and G. B. Giannakis, "Optimal and Scalable Caching for 5G Using Reinforcement Learning of Space-Time Popularities," *IEEE J. Sel. Areas Signal Process.*, vol. 12, no. 1, pp. 180–190, Feb. 2018.

[23] D. Applegate, A. Archer, V. Gopalakrishnan, S. Lee, and K. K. Ramakrishnan, "Optimal Content Placement for a Large-Scale VoD System," *IEEE/ACM Trans Netw.*, vol. 24, no. 4, pp. 2114–2127, Aug. 2016.

[24] B. N. Bharath, K. G. Nagananda, D. Gündüz, and H. V. Poor, "Caching With Time-Varying Popularity Profiles: A Learning-Theoretic Perspective," *IEEE Trans. Commun.*, vol. 66, no. 9, pp. 3837–3847, Sept. 2018.

[25] J. Gao, L. Zhao, and X. Shen, "The Study of Caching via State Transition Field - the Case of Time-Invariant Popularity", *IEEE Trans. Wireless Commun.*, accepted.

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/TWC.2019.2940667, IEEE Transactions on Wireless Communications



Jie Gao (S'09-13, M'17) received the B.Eng. degree in electronics and information engineering from Huazhong University of Science and Technology, Wuhan, China, in 2007, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Alberta, Edmonton, Alberta, Canada, in 2009 and 2014, respectively. He is a recipient of Alberta Innovates -Technology Futures scholarship, Ontario Centres of Excellence TalentEdge Fellowship, and Natural Science and Engineering Research Council of Canada Postdoctoral Fellowship. He is currently a research associate with the Department of Electrical & Computer Engineering at University of Waterloo. His research interests include the application of game theory, mechanism design, and optimization methods for distributed decision making and network utility maximization in wireless communication networks.

Lian Zhao (S'99-M'03-SM'06) received the Ph.D. degree from the Department of Electrical and Computer Engineering (ELCE), University of Waterloo, Canada, in 2002. She joined the Department of Electrical and Computer Engineering at Ryerson University, Toronto, Canada, in 2003 and be a Professor in 2014. Her research interests are in the areas of wireless communications, resource management, mobile edge computing, communication and caching, network slicing and scalable network control.

She received the Best Land Transportation Paper Award from IEEE Vehicular Technology Society in 2016; Top 15 Editor in 2015 for IEEE Transaction on Vehicular Technology; Best Paper Award from the 2013 International Conference on Wireless Communications and Signal Processing (WCSP) and Best Student Paper Award (with her student) from Chinacom in 2011; the Canada Foundation for Innovation (CFI) New Opportunity Research Award in 2005, and Early Tenure and promotion to Associate Professor in 2006. She serves the editorial board for IEEE Transaction on Vehicular Technology, IEEE Internet of Things Journal, Transactions on Emerging Telecommunication Technologies; General co-Chair for IEEE GreeenCom 2018; co-chair for IEEE ICC 2018 Wireless Communication Symposium; workshop co-chair for IEEE/CIC ICCC 2015; co-chair for IEEE Global Communications Conference (GLOBECOM) 2013 Communication Theory Symposium. She served as a committee member for NSERC (Natural Science and Engineering Research Council of Canada) Discovery Grants Evaluation Group for Electrical and Computer Engineering 2015 to 2018. She is a licensed Professional Engineer in the Province of Ontario, a senior member of the IEEE Communication and Vehicular Society.

Xuemin (Sherman) Shen (IEEE M'97-SM'02-F'09) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research focuses on resource management, wireless network security, social networks, 5G and beyond, and vehicular ad hoc and sensor networks. He is a registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society.

Dr. Shen received the R.A. Fessenden Award in 2019 from IEEE, Canada, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, the Joseph LoCicero Award in 2015 and the Education Award in 2017 from the IEEE Communications Society. He has also received the Excellent Graduate Supervision Award in 2006 and the Outstanding Performance Award 5 times from the University of Waterloo and the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee Chair/Co-Chair for the IEEE Globecom'16, the IEEE Infocom'14, the IEEE VTC'10 Fall, the IEEE Globecom'07, the Symposia Chair for the IEEE ICC'10, the Tutorial Chair for the IEEE VTC'11 Spring, the Chair for the IEEE Communications Society Technical Committee on Wireless Communications, and P2P Communications and Networking. He is the Editor-in-Chief of the IEEE INTERNET OF THINGS JOURNAL and the Vice President on Publications of the IEEE Communications Society.