2021 19th International Symposium on Modeling and Optimization in Mobile, Ad hoc, and Wireless Networks (WiOpt)

A Coupon Collector based approximation for LRU cache hits under Zipf requests

Pawan Poojary
ECE Department
Northwestern University
Evanston, IL 60208, USA.
Email: pawanpoojary2018@u.northwestern.edu

Sharayu Moharir
Department of Electrical Engineering
IIT Bombay
Mumbai 400076, India.
Email: sharayum@ee.iitb.ac.in

Krishna Jagannathan
Department of Electrical Engineering
IIT Madras
Chennai 600036, India.
Email: krishnaj@ee.iitm.ac.in

***Abstract*—The Least Recently Used (LRU) policy is widely used in caching, since it is computationally inexpensive and can be implemented 'on-the-fly.' However, existing analyses of content-wise hit-rates under LRU have expressions whose complexity grows rapidly with the buffer size. In this paper, we derive a simple yet accurate approximation for the LRU content-wise hit-rates under Zipf-distributed requests, in the regime of a large content population. To this end, we map the characteristic time of a content in the LRU policy to the classical Coupon Collector's Problem (CCP). We justify the accuracy of these approximations by showing analytically that the characteristic time concentrates sharply around its mean. Our bounds highlight and quantify the impact of cache-size scaling as well as the variations in content popularity on the accuracy of the hit-rate estimates. Specifically, we show that these estimates become more accurate with a decrease in Zipf parameter β or an increase in the cache-size scaling. Finally, our analysis of the CCP with Zipf-distributed coupons could be of independent interest.**

***Index Terms*—Sensor networks, hit-rate, Zipf's law, LRU, characteristic time, Coupon Collector's Problem (CCP), Chernoff bound.**

I. Introduction

DISTRIBUTED content caching has been employed in networks on a large scale owing to its well-known advantages [1]. Caching contents close to the end-users in a network serves to minimize the load on the network back-end by transferring it towards the network end-nodes. Moreover, the reduction in the back-end traffic saves precious network bandwidth and decreases content-delivery latency. However, owing to physical resource constraints, caches are capable of storing only a small fraction of the entire content catalogue. This is further exacerbated by the ever-increasing content population. Hence, optimal cache replacement strategies aimed at minimizing the number of deferred requests must be devised.

It is a well-established fact in literature that, given a fixed cache-size, caching the most popular contents is optimal [2]. However, this optimality is achieved at the cost of a large memory[1] (equal to number of content types). This memory requirement arises from the need to estimate content popularities from the incoming content requests so that contents with higher popularity can be accomodated in the cache. Least Frequently Used (LFU) being the most basic among such policies, replaces cached contents based on frequency measurements of past requests[2]. On the other hand, the Least Recently Used (LRU) policy, while sub-optimal [7], is widely used owing to certain desirable properties. Specifically, LRU does not require the popularity estimates of content requests, and is hence computationally inexpensive. Also, it requires a much smaller memory, equal in size to the cache.

The probability that a particular content is available in the cache upon request, referred to as that content's *hit-rate*, is an important quantity in understanding the caching performance. Exact hit-rates for the LRU policy were derived in [7] and [8]; however the complexity involved in computing these hit-rates grows exponentially in the number of objects and cache-size. As a result, finding approximate hit-rate expressions that have significantly lower computational complexity while maintaining reasonable accuracy has received significant attention in the literature [9]–[11]. Moreover, these performance estimates have been used in the recent literature to simplify the analyses of complex hierarchical caching topologies [12].

In this paper, we propose an accurate approximation for the content-wise hit-rates of LRU policy under Zipf distributed requests, in the regime of a large object population. Our analysis is based on relating the *characteristic time* [12] of a content in the LRU policy to the classical Coupon Collector's Problem (CCP) [13]. We show analytically that the characteristic time enjoys a tight concentration about its mean, thereby justifying as well as quantifying the accuracy of such approximations. Further, we show that the accuracy of the hit-rate estimate improves with a decrease in the Zipf parameter β or an increase in the cache-size scaling.

We believe that our analysis of the CCP with Zipf-distributed coupons is of independent interest, since the CCP is a classical problem with applications in several areas of engineering, linguistics and biology.

A. Related Work

Several studies, using empirical data from traces of web proxy caches, have shown that content request distributions strongly follow the Zipf's law with varying exponents [14],

[1]Memory refers to the amount of resources required for the execution of these caching algorithms, *i.e.*, their space complexity.

[2]Several methods have been proposed in the literature to estimate the content popularity at lower costs [3]–[6].

ISBN 978-3-903176-37-9 © 2021 IFIP

[15]. Works by [6] and [16] exploit this Zipfian nature of requests to propose design rules for the sizing of web caches that implement LFU policy to achieve the desired hit-rate requirements. Our study, on the other hand, focuses on providing accurate hit-rate estimates under Zipf requests for the much simpler to implement LRU policy in a cache of fixed size. Subsequent to the analyses by [7] and [8] that provided exact hit-rates for the LRU policy, Dan *et al.* [9] obtained approximate hit-rates with much lower computational complexity.

Fagin [10] introduced a useful approximation for LRU under the independent reference model (IRM) which was that in the asymptotic sense, LRU hit-rate converges to that of a time-to-live (TTL) cache with its timer set to a time for which the expected working set size equals the cache-size. Che *et al.* [12] furthered this work by proposing a simple approach to estimate LRU hit-rates using certain approximations and by coming up with the notion of a *characteristic time* of the cache. The accuracy of the hit-rates and the scope of this approximation were further investigated in [17]. Our approximate hit-rates could be considered as a discrete time equivalent of Che's approximation given in [12], [17] with results that prove asymptotic accuracy and provide the associated convergence rates under Zipf requests. Similar to our approach, the expressions in [10], [12], [17] also relate to the expected waiting time for the CCP. However, unlike these prior works, along with providing approximate hit-rates for the LRU policy under Zipf requests, we also provide concentration bounds on the waiting time, which serve to quantify the accuracy of these approximations.

More recently, a closely related work [18] extends the LRU approximation beyond the IRM to multiple content request flows that form independent stationary and ergodic processes. They prove asymptotic accuracy given a fixed cache-size scaling for a large class of popularity distributions. However, owing to this generality in distribution, the effect of varying the cache-size scaling on the convergence rate of the hit-rate estimate is not well understood. Whereas our work, although being restricted to power law (Zipf) popularities, explicitly shows how varying the fractional cache-size and the power law exponent affect the convergence rate.

An interesting work by [19] provides asymptotic hit-rates when multiple flows of requests for varying data item sizes are served by a shared LRU cache space. It considers requests belonging to a broad class of power-law distributions (including Zipf and Weibull) and presents conditions when cache space pooling is preferred over splitting. Another work by [20] considers requests served by an LRU cache for contents that are divided into two classes, with uniform popularity within each class; we instead consider power law popularities that seem more realistic from empirical evidence.

Studies in [21]–[26] address the notion of freshness by considering that a content entering the cache has a limited lifetime beyond which it expires, i.e., it is no longer relevant to the end-user as a new updated version of that content is available in the back-end. Whereas our model does not consider version updates and the same content is available from the source upon request. In contrast to [21] which considers infinite cache, we obtain hit-rates under finite caching resources. Lastly, the work by [27] derives approximate LRU hit-rate, but does not provide justification for the approximation. Our work, in addition, provides insights on the regimes where the approximation is justified.

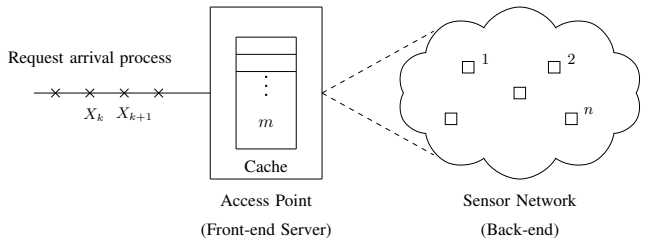


Fig. 1: Content delivery to incoming requests through a front-end server equipped with finite caching resources.

B. Organisation

The remainder of the paper is organized as follows. Firstly, we describe our system model in Section II. Metrics for evaluation of cache performance are defined and the policy that achieves optimal performance is described in Section III. We analyse hit-rates under the LRU policy and propose hit-rate estimates for Zipf distributed requests in Section IV. Analytical arguments that establish the accuracy of these hit-rate estimates are provided in Section V. We present the simulation results in Section VI. Proofs outlines of the main results obtained in this paper are provided in Section VII, and the detailed proofs are deferred to [28]. We present our conclusions in Section VIII.

II. System Model

In our model depicted in Figure 1, we consider a front-end server with a finite cache. It serves incoming client requests with contents fetched from a back-end.

A. Server and storage model

The system consists of an Access Point (AP) equipped with a cache of length m. The AP fetches contents from a population of n content-generating objects indexed by $\{1, 2, \ldots, n\}$ and stores them in the cache in order to serve future requests. The object population could constitute a sensor grid where several smart-devices connect wirelessly to the AP. Each content fetched from the objects is of unit size and occupies unit space in the cache. Owing to practical resource constraints, typically $m \ll n$. The AP acts as a front-end server and is assigned the task of serving incoming content requests either from its cache or by fetching contents from the back-end.

B. Request arrival model

We adopt the Independent Reference Model (IRM) which is known to be a well suited abstraction for independent requests generated from a large population of users [14], [7], [29].

ISBN 978-3-903176-37-9 © 2021 IFIP

The request arrival process is modeled as an infinite sequence of independent and identically distributed random variables $\{X_1, X_2, \ldots\}$, where $X_k = i$ denotes that the k^{th} request is for content type $i \in \{1, 2, \ldots, n\}$. We denote popularity of content i by $p_i \triangleq \mathbb{P}(X_k = i)$ and assume that the content requests are Zipf distributed. Under the Zipf's law, $p_i \propto i^{-\beta}$, where the exponent $\beta \geq 0$ is known as the Zipf parameter[3].

C. Service model

The server serves the incoming requests in the following manner:

- If the content corresponding to a request is not present in the cache, a cache *miss* occurs.
- If the requested content is present in the cache, a cache *hit* occurs and the request is served.
- In the event of a cache miss, the server has to fetch the content from the back-end and serve the request. At this point, if the cache is full, the server has to decide whether or not to replace an existing content in the cache with this fetched content. This strategy is referred to as the replacement or caching policy.

Remark 1. A content fetch is initiated if and only if there is a cache miss.

III. System Performance Metrics

The hit-rate or hit-ratio of object i at time t, $H_{\mathcal{A}}(i,t)$ is the ratio of the number of cache hits to the total number of requests for object i received till time t under a policy $\mathcal{A} \in \mathbb{A}$. Here, $\mathbb{A}$ is the set of all replacement policies. The steady-state hit-rate $h_{\mathcal{A}}(i) = \lim_{t\to\infty} H_{\mathcal{A}}(i,t)$[4] is the steady-state probability that content i is present in the cache, simply referred to as the *hit rate* of object i under policy $\mathcal{A}$. This follows from the fact that the caching process is ergodic[5].

However, the overall cache performance is quantified by the total probability of a cache hit in the steady state, simply referred to as the *hit probability*. Let Y_k^i denote the event that content i is present in the cache in the k^{th} time-slot. Then, the hit probability in a discrete time slot for a policy $\mathcal{A} \in \mathbb{A}$ is given by

$$\begin{aligned}\mathbb{P}_{\mathcal{A}}(\text{hit}) &\triangleq \sum_{i=1}^{n} \mathbb{P}\big(\textit{cache hit} \mid X_k = i\big)\, \mathbb{P}(X_k = i) \\ &\overset{(a)}{=} \sum_{i=1}^{n} \mathbb{P}\big(Y_k^i \mid X_k = i\big)\, \mathbb{P}(X_k = i) \\ &\overset{(b)}{=} \sum_{i=1}^{n} \mathbb{P}(Y_k^i)\, \mathbb{P}(X_k = i) = \sum_{i=1}^{n} h_{\mathcal{A}}(i)\, p_i. \qquad (1)\end{aligned}$$

Step (a) holds due to the fact that, under the condition that content i is requested, a cache hit is analogous to content i being present in the cache. Step (b) follows from the fact that, in the k^{th} time-slot, the presence of a content in the cache is independent of any ensuing request for it.

Now, given a finite cache of size m, let $\mathbb{I}_{\{i\in\text{cache}\}}$ denote the indicator r.v. that indicates the presence of content i in the cache in steady-state, under a policy $\mathcal{A}$. Hence, $\mathbb{P}\big(\mathbb{I}_{\{i\in\text{cache}\}} = 1\big) = h_{\mathcal{A}}(i)$. Further, as there can be atmost m fresh contents in the cache at any given time, it follows that

$$\sum_{i=1}^{n} \mathbb{I}_{\{i\in\text{cache}\}} \leq m \;\Rightarrow\; \sum_{i=1}^{n} h_{\mathcal{A}}(i) \leq m, \;\; \forall\, \mathcal{A} \in \mathbb{A}. \qquad (2)$$

Without loss of generality, let the n objects be indexed in the decreasing order of their popularities, *i.e*, $p_1 \geq p_2 \geq \ldots \geq p_n$. Then, subject to the constraint given by inequality (2), the optimal policy would be the one to achieve the hit-rates: $h^*(i) = 1$, for $i \in \{1, 2, \ldots, m\}$, and 0 otherwise. Simply put, the optimal policy is to only retain the m most popular contents in the cache, as has been stated in the literature.

IV. Hit-rate analysis of the LRU Policy

The LRU policy replaces the least recently requested content from the cache with the fetched content. Let $\{X_1, X_2, \ldots\}$ be the request stream, C_i refer to content i and LUT(C_i) refer to its last used time-slot. Then, the LRU policy is implemented as per Algorithm 1.

Algorithm 1 LRU policy implementation.

```
k ← 1.
loop
    if C_{X_k} ∈ cache then                          ▷ cache hit
        serve the request
    else                                             ▷ cache miss
        Fetch content type C_{X_k} from back-end
        Serve the request
        Replace C_{LRU} with C_{X_k}; where
        LRU ≜ arg min_i {LUT(C_i) : C_i ∈ cache}
    end if
    LUT(C_{X_k}) ← k
    k ← k + 1
end loop
```

Now, the hit-rate of any content $i \in \{1, 2, \ldots, n\}$ is closely related to its sojourn time in the cache. Once content i arrives in the cache upon a request (clearly due to a cache miss), it could subsequently be evicted out of the cache before being requested again. This time duration spent in the cache is of particular interest for obtaining the hit-rates and is defined as follows:

A. Characteristic time of content i, $T_c(i)$

Denote by $T_c(i)$, the number of time slots by which m distinct contents other than C_i are requested at least once. We refer to $T_c(i)$ as the *characteristic time* of content i.

[3]Typical values for β reported by empirical studies conducted on a variety of networks are: 0.64 to 0.83 [14] and 1 to 2.07 [30].

[4]The limit exists since the caching process constitutes an ergodic Markov chain, that is, it is irreducible and aperiodic.

[5]A process whose time average converges to its ensemble average.

ISBN 978-3-903176-37-9 © 2021 IFIP

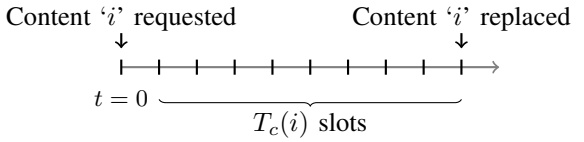


Fig. 2: It takes $T_c(i)$ time slots for content i to become the least recently used amongst cached contents.

Suppose that C_i is requested in the present slot ($t = 0$). Assuming that the contents in the cache are ordered as most recently used first, C_i currently occupies the top position. In the ensuing time, it takes m distinct requests apart from C_i for C_i to shuffle to the bottom of the cache and get evicted. This is true provided C_i is not requested again before getting evicted. Under this condition, $T_c(i)$ denotes the number of time slots for which C_i stays in the cache before getting replaced. Hence, quantifying $T_c(i)$ becomes important for calculating the hit-rate of C_i under the LRU policy. The characteristic time $T_c(i)$ can be mapped to the celebrated *Coupon Collector's Problem* described below.

Coupon Collector's Problem (CCP): Given a collection $\mathcal{C} = \{1, 2, \ldots, n\}$ of n coupons with p_i being the probability of drawing coupon i, determine the number of independent draws (with replacement) from $\mathcal{C}$ to first obtain a collection of m different coupons.

In context of the above problem, let T_m denote the required number of draws referred to as the *waiting time* for the Coupon Collector's Problem. The expected number of draws has been shown to satisfy the following equation in [8]:

$$\mathbb{E}(T_m) = \sum_{q=0}^{m-1} (-1)^{m-1-q} \binom{n-q-1}{n-m} \sum_{|J|=q} \frac{1}{1-P_J}, \quad (3)$$

where $P_J = \sum_{i \in J} p_i$.

In the context of the LRU policy with a cache of size m, $T_c(i)$ is analogous to the waiting time T_m for the Coupon Collector's Problem. In particular, the request for a content C_k in the former corresponds to a coupon k being drawn from the collection $\mathcal{C}$ in the latter. However, the only difference is that unlike T_m, $T_c(i)$ denotes the number of coupons drawn until the m^{th} distinct coupon *excluding* i is observed. That is, the drawing of coupon i is equivalent to a blank draw. We now make approximations for calculating $T_c(i)$ which are similar in spirit to those of Che *et al.* [12] and [17].

B. Approximations for calculating $T_c(i)$

Let $\overline{T}_c$ denote the number of time-slots until $m+1$ distinct contents are requested. Let X_1 denote the first request. Now, if the first request is for content i, *i.e.*, $X_1 = i$, then $T_c(i) = \overline{T}_c - 1$. Using this as the basis yields the following result:

$$\mathbb{E}(\overline{T}_c) = 1 + \sum_{i=1}^{n} \mathbb{E}\big(T_c(i)\big)\, \mathbb{P}(X_1 = i). \quad (4)$$

Approximation 1: The dependence of $\mathbb{E}(T_c(i))$ on i can be ignored, *i.e.*, $\mathbb{E}(T_c(i)) \approx t_c \; \forall \; i$.

This is a reasonable and widely-used [12], [17] approximation when the individual popularities are relatively insignificant to their sum, and becomes exact if the requests are equiprobable. We support this claim for Zipf distributed requests using numerical simulations provided in Sub-section VI-A. We notice that the error in approximating $\mathbb{E}(T_c(i))$ with t_c reduces with decreasing values of β, with increasing values of the $\frac{m}{n}$ ratio and also as $n \to \infty$ for a fixed $\frac{m}{n}$ ratio. Using this approximation in (4), we get

$$\mathbb{E}(T_c(i)) \approx t_c \triangleq \mathbb{E}(\overline{T}_c) - 1, \;\; \forall \, i \in \{1, 2, \ldots, n\}. \quad (5)$$

Note that Approximation 1 could be widely off in the setting where one or a small number of contents correspond to a large fraction of requests.

Approximation 2: We assume that for large n, the random variable $T_c(i)$ is well approximated by its expected value (*i.e.*, it is nearly deterministic).

In Section V-B, we provide analytical justifications for the above approximation under Zipf distributed requests with parameter $\beta \in [0, \infty)$. Using this approximation in equation (5), we get $T_c(i) \approx t_c, \quad \forall \; i \in \{1, 2, \ldots, n\}$. In the works by [12] and [17], t_c is referred to as the characteristic time of the *cache*. Finally, from the definition of $\overline{T_c}$, it follows that $\overline{T_c} = T_{m+1}$. Hence, $\mathbb{E}(\overline{T_c})$ can be calculated from equation (3), which is then used in equation (5) to compute t_c.

C. Hit-rate for LRU policy

As depicted in Figure 2, the duration for which content i resides in the cache, since its last request, is limited by its characteristic time $T_c(i)$. Let τ_i denote the inter-arrival time between requests for content i under the IRM; $\tau_i \sim Geom(p_i)$. For a content i requested in the present slot, a request after τ_i slots will result in a cache hit if and only if $\tau_i < T_c(i)$. Therefore, the hit-rate is given by $h_{LRU}(i) = \mathbb{P}(\tau_i < T_c(i)) \approx \mathbb{P}(\tau_i < t_c)$ after applying the approximations from Section IV-B. This yields the following aproximate hit-rate expression for content $i \in \{1, 2, \ldots, n\}$ under the LRU policy.

$$h_{LRU}(i) \approx 1 - (1 - p_i)^{t_c - 1}. \quad (6)$$

And the hit probability denoted by $P_{LRU}(hit)$ can then be obtained using the above estimates in (1).

In the subsequent sections, we provide analytical results as well as illustrate using numerical simulations that: under Zipf distributed requests and for a fairly large object population relative to the size of the cache, the above approximation is generally quite accurate.

V. Asymptotic Analysis of Characteristic Time under Zipf Requests

In this section, we provide analytical results to justify our approximation that $T_c(i)$ is nearly deterministic for large n under Zipf distributed requests (Approximation 2). As mentioned earlier, the characteristic time $T_c(i)$ of the LRU policy is analogous to the waiting time T_m for the Coupon Collector's Problem. Hence, in the rest of this section, we use the parlance of the Coupon Collector's Problem. Since

ISBN 978-3-903176-37-9 © 2021 IFIP

the asymptotic behaviour of T_m and $T_c(i)$ are identical, we characterize T_m instead of $T_c(i)$ to simplify the analysis.

A. Limit Theorems for the Convergence of T_m to m

In the following, we provide a limit theorem for the case where the coupon draws obey the Zipf's law with Zipf parameter $\beta \in [0, 1)$. Note that the Zipf's law generalises the uniform coupon draw case for which, the asymptotic distribution for the waiting time has been investigated in the works by [31], [32] and [33].

Theorem 1. *Consider the coupon draws being sampled from a Zipf distribution with parameter $\beta \in [0, 1)$, such that the n coupons are indexed in the decreasing order of popularity. If $m = o\left(n^{\frac{1-\beta}{2-\beta}}\right)$, then $T_m \overset{i.p}{\to} m$.*

In context of the LRU policy, the above theorem implies that if the cache-size m scales slower than $n^{\frac{1-\beta}{2-\beta}}$, then asymptotically, $T_c(i)$ converges to m in probability. Refer to Section VII for a proof outline. The study by [32] provided limiting distributions for T_m depending on how m scales with n as $n \to \infty$. In particular, for the uniform case, they derived sufficient conditions on the scaling of m under which $T_m \overset{i.p}{\to} m$. This result becomes a corollary to Theorem 1 for $\beta = 0$ and is stated below.

Corollary 1.1. *Consider the coupon draws being equiprobable. If $m = o(\sqrt{n})$, then $T_m \overset{i.p}{\to} m$.*

In addition to justifying Approximation 2, Theorem 1 also provides a much simpler approximation to T_m that could simplify the analysis of LRU caching in inter-connected cache networks. However, the result is restricted to the regime $\beta < 1$ and $m = o\left(n^{\frac{1-\beta}{2-\beta}}\right)$.

B. Concentration Bounds on the Deviations of T_m from its Expected Value

In this section, we derive tail bounds for the δ-deviations of $\frac{T_m}{\mathbb{E}(T_m)}$ about unity. Further, we show that these concentration bounds are asymptotically tight in n thereby implying that $\frac{T_m}{\mathbb{E}(T_m)}$ converges in probability to 1 as $n \to \infty$. This analytically validates our assumption that for large n, the random variable $T_c(i)$ is well approximated by its expected value.

Note that the qualitative behaviour of Zipf distribution as $n \to \infty$ varies with the value of the Zipf parameter β. More specifically, given the Zipf distribution with $p_i \propto i^{-\beta}$, the normalizing factor $\sum_{i=1}^{n} i^{-\beta}$ known as the Riemann's zeta function $\zeta(\beta)$ is finite only if $\beta > 1$. Hence, we conduct separate analyses for three regions of the Zipf parameter space, namely; $\beta < 1$ (which includes $\beta = 0$ (uniform) as a special case), $\beta = 1$ and $\beta > 1$. We present the results obtained from these analyses in the form of separate theorems as follows.

1) $\beta < 1$:

Theorem 2. *Consider the coupon draws being Zipf distributed with $\beta < 1$. If $m = o(n)$, then for any $\delta > 0$,*

$$\mathbb{P}\left(\frac{T_m}{\mathbb{E}(T_m)} < (1-\delta)\right) \leq \exp\left(-\frac{m}{16}\,\delta^2\right)$$
$$\text{and } \mathbb{P}\left(\frac{T_m}{\mathbb{E}(T_m)} > (1+\delta)\right) \leq \exp\left(-n^{\frac{1-\beta}{2}}\, m^{\frac{1+\beta}{2}}\, \delta^{\frac{3}{2}}\right).$$

Corollary 2.1. *Consider the coupon draws being equiprobable. If $m = o(n)$, then for any $\delta > 0$,*

$$\mathbb{P}\left(\frac{T_m}{\mathbb{E}(T_m)} < (1-\delta)\right) \leq \exp\left(-\frac{m}{16}\,\delta^2\right)$$
$$\text{and } \mathbb{P}\left(\frac{T_m}{\mathbb{E}(T_m)} > (1+\delta)\right) \leq \exp\left(-\sqrt{nm}\,\delta^{\frac{3}{2}}\right).$$

2) $\beta = 1$:

Theorem 3. *Consider the coupon draws being Zipf distributed with $\beta = 1$. If $\ln m = o(\ln n)$, then for any $\delta > 0$,*

$$\mathbb{P}\left(\frac{T_m}{\mathbb{E}(T_m)} < (1-\delta)\right) \leq \exp\left(-\frac{m}{16}\,\delta^2\right)$$
$$\text{and } \mathbb{P}\left(\frac{T_m}{\mathbb{E}(T_m)} > (1+\delta)\right) \leq \exp\left(-m\sqrt{\frac{\ln n}{\ln m}}\,\delta^{\frac{3}{2}}\right).$$

3) $\beta > 1$:

Theorem 4. *Consider the coupon draws being Zipf distributed with $\beta > 1$. If $m = o(n)$, then for any $\delta > 0$,*

$$\mathbb{P}\left(\frac{T_m}{\mathbb{E}(T_m)} < (1-\delta)\right) \leq \exp\left(-\frac{m}{8}\,\delta^2\right)$$
$$\text{and } \mathbb{P}\left(\frac{T_m}{\mathbb{E}(T_m)} > (1+\delta)\right) \leq \exp\left(-n^{\frac{1-\beta}{2}}\, m^{\frac{1+\beta}{2}}\, \delta\right).$$

For proving the above theorems, we first express the waiting time as $T_m = \sum_{k=1}^{m} N_k$ where N_k denotes the number of draws to obtain the k^{th} distinct coupon having obtained a set of $k-1$ distinct coupons. Next, we use the *Chernoff* bound technique for the sum of independent r.v's on T_m which in turn requires the m.g.f of N_k, $\mathbb{E}(e^{\lambda N_k})$. This m.g.f has a simple expression for $\beta = 0$. However, for $\beta \neq 0$, the m.g.f has to be expressed combinatorially. In order to circumvent this issue, we apply conditional m.g.f arguments to appropriately bound the m.g.f. of N_k. A proof outline of Theorems 2 and 3 is provided in Section VII. The proof for Theorem 4 has the same underlying intuition and proceeds mostly in a similar manner. We provide the detailed proofs in [28]. At this point, we state an important result which directly follows from Theorems 2, 3 and 4.

Theorem 5. *Consider the coupon draws being sampled from a Zipf distribution with parameter $\beta \in [0, \infty)$, such that the n coupons are indexed in the decreasing order of popularity. Then, $\frac{T_m}{\mathbb{E}(T_m)} \overset{i.p}{\to} 1$ under the following sufficient conditions:*

ISBN 978-3-903176-37-9 © 2021 IFIP

- *If* $m = \omega(1)$ *and* $m = o(n)$, *for* $\beta < 1$,
- *If* $m = \omega(1)$ *and* $\ln m = o(\ln n)$, *for* $\beta = 1$, *and*
- *If* $m = \omega(n^{\frac{\beta-1}{\beta+1}})$ *and* $m = o(n)$, *for* $\beta > 1$.

In context of the LRU policy, for the case of Zipf distributed requests with $\beta < 1$, the above theorem implies that Approximation 2 is justified for any sub-linear scaling of the cache-size. Whereas for $\beta > 1$, the cache-size with a scaling which is atleast faster than $n^{\frac{\beta-1}{\beta+1}}$ while still being sub-linear in n guarantees the validity of Approximation 2. Note that the condition: $m = \omega(1)$ and $m = o(n)$ is a reasonable assumption in practice and does not impose any significant restrictions on caching systems. The reason being that, in order to enhance the performance, the cache-size has to be increased in response to the growing object population. However, this scaling cannot be achieved in a linear fashion due to physical resource constraints.

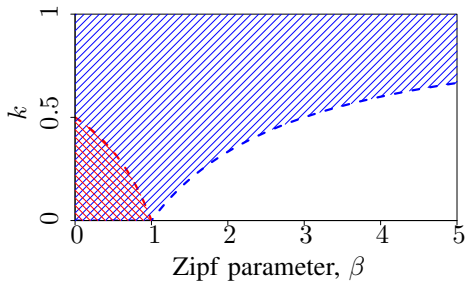


Fig. 3: The blue shaded region denotes the sufficient condition to be satisfied by (k, β) *such that for* $m = \omega(n^k)$, $\frac{T_m}{\mathbb{E}(T_m)} \xrightarrow{i.p} 1$. *The red shaded region denotes the sufficient condition to be satisfied by* (k, β) *such that for* $m = o(n^k), T_m \xrightarrow{i.p} m$.

The results obtained in Theorems 1 and 5 are summarized in Figure 3. Note that $T_m \xrightarrow{i.p} m$ is a stronger condition than $\frac{T_m}{\mathbb{E}(T_m)} \xrightarrow{i.p} 1$ for Approximation 2 to hold. Hence, for the case of $\beta < 1$, it is evident from the shaded regions in Figure 3 that, in comparison to Theorem 1, Theorem 5 provides a much milder condition on the cache-size scaling, thereby justifying the approximation for a wider class of practical scenarios.

Lastly, it is evident from the expressions for the δ-deviation bounds obtained in the above theorems that the rate of convergence of these bounds is influenced by two factors, namely, the Zipf-parameter β and the cache-size scaling. Correspondingly, the following two trends in the rate of convergence of the concentration bounds are observed. First, the rate of convergence becomes slower with an increase in Zipf parameter β. Second, the rate of convergence becomes faster with an improvement in the cache-size scaling. In conclusion, the approximation of $T_c(i)$ by its expected value becomes better with a decrease in β and an increase in the cache-size scaling.

In the next section, we empirically show that the same variations in the system parametrs, *i.e.,* a decrease in β and an increase in the cache-size scaling also make Approximation 1 better. Further, we demonstrate through simulations that by using both Approximations 1 and 2, the hit-rate estimates obtained in (6) are reasonably accurate.

VI. Simulation Results

A. Simulations to validate Approximation 1

We first obtain $\mathbb{E}(T_c(i))$ for all $i \in \{1, 2, \ldots, n\}$ and $\mathbb{E}(\overline{T}_c)$ by averaging these quantities over sufficiently large number of simulations. We then use $\mathbb{E}(\overline{T}_c)$ and obtain t_c from equation (5) to calculate the error in approximating $\mathbb{E}(T_c(i))$. This error denoted by μ is given by $\mu = \frac{|\mathbb{E}(T_c(i)) - t_c|}{\mathbb{E}(T_c(i))}$.

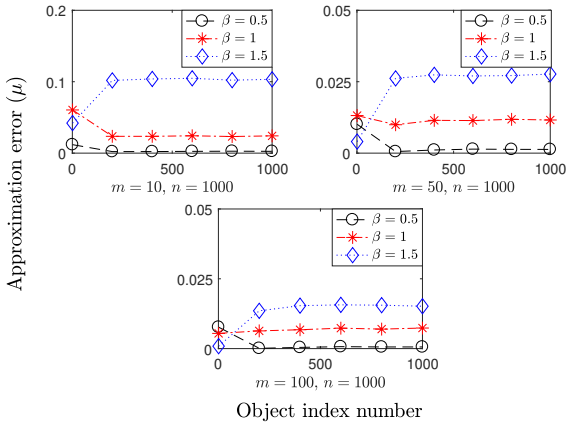


Fig. 4: Error in approximating $\mathbb{E}(T_c(i))$ *against the object index* i *for values of Zipf parameter* $\beta = 0.5, 1, 1.5$.

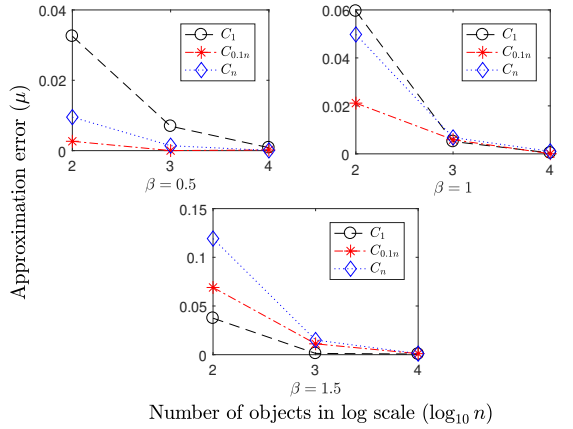
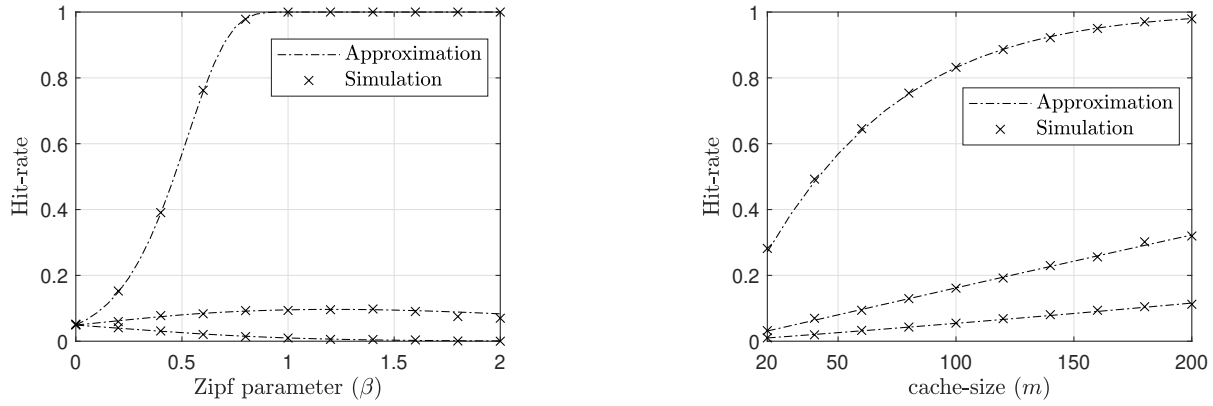


Fig. 5: Asymptotic trend for the error in approximating $\mathbb{E}(T_c(i))$ *as the number of objects,* $n \to \infty$. *Here, cache-size* $m = 0.1n$ *and* $\beta = 0.5, 1, 1.5$.

In Figure 4, we plot the error against content index for a total number of contents $n = 1000$. We consider different $\frac{m}{n}$ ratios and different values of β to obtain the plots. We notice that the error in approximating $\mathbb{E}(T_c(i))$ with t_c reduces with decreasing values of β and increasing values of the $\frac{m}{n}$ ratio.

ISBN 978-3-903176-37-9 © 2021 IFIP



(a) *(b)*

Fig. 6: Hit-rate against (a) Zipf parameter β for contents 1, 100 and 1000; $n = 1000$, $m = 50$ and (b) cache-size m for contents 1, 100 and 1000; $n = 1000$ and $\beta = 0.5$. Content with lower index has higher hit-rate.

Further, in Figure 5, we plot the error against the number of objects n for contents with indices $1, 0.1n$ and n denoted by $C_1, C_{0.1n}$ and C_n respectively. We fix values for β and $\frac{m}{n}$ and observe that the error decays to zero as $n \to \infty$. From this, we infer that Approximation 1 becomes better with an increase in the number of objects.

B. Accuracy of proposed LRU hit-rates under Zipf distributed requests

We depict the accuracy of the approximate LRU hit-rates obtained in (6) for Zipf distributed requests over a wide range of system parameters. The plots shown in Figures $6(a)$ and $6(b)$ correspond to hit-rate variations with respect to β and m respectively for selected content indices: 1, 100 and 1000. Hit-rates are obtained by simulating the caching process for sufficiently long runs to ensure their high accuracy. Whereas, approximate hit-rates are obtained from (6) which requires the value of $\mathbb{E}(\overline{T_c})$ to be calculated from the expression for the mean waiting time given in (3). Note that, although this computation has an exponential complexity in n, we use the methods provided by [8] and [34] wherein $\mathbb{E}(\overline{T_c})$ can be computed with high accuracy in a time that is linear in n. We do not elaborate on this method as efficient computation of the waiting time is not the main focus of this paper. We infer from Figure 6 that the theoretical approximation for the LRU hit-rates provided in (6) matches the simulations reasonably well.

VII. Proof of main results

We present proof outlines for Theorems 1, 2 and 3 and defer the detailed proofs to [28].

A. Proof Outline of Theorem 1

1. We express the waiting time as $T_m = \sum_{k=1}^{m} N_k$ where N_k denotes the number of draws to obtain the k^{th} distinct coupon having obtained a set of $k-1$ distinct coupons.
2. To prove that $T_m \overset{i.p}{\to} m$, it is sufficient to show that $\mathbb{P}(T_m = m) \to 1$. To do this, we obtain a lower bound to $\mathbb{P}(T_m = m)$ and show that it converges to 1.
3. To get this lower bound, we use the fact that $T_m = m$ only if $N_k = 1$ for all k. We then find a lower bound for $\mathbb{P}(N_k = 1)$ which results in a lower bound for $\mathbb{P}(T_m = m)$.

B. Proof Outline of Theorems 2 and 3

1. We have $T_m = \sum_{k=1}^{m} N_k$ as before (refer to the proof of Theorem 1). To obtain the lower tail bound, we apply Chernoff's bound on T_m for a $\delta-$deviation from its mean which gives $\mathbb{P}\big(T_m < \mathbb{E}(T_m)(1-\delta)\big) \leq \exp\big(\inf_{\lambda>0} f(\lambda)\big)$. Here, $f(\lambda) = d\lambda^2 - c\lambda$ with constants $c, d > 0$, which is minimized by $\lambda_{\min} = \frac{c}{2d}$. This yields the required bound on further simplification.
2. Similarly, we obtain an upper tail bound as $\mathbb{P}\big(T_m > \mathbb{E}(T_m)(1+\delta)\big) \leq \exp\big(\inf_{\lambda>0} g(\lambda)\big)$. Here, $g(\lambda) = -a\lambda + (e^{\lambda} - 1)b$ with constants $a, b > 0$, which is minimized by $\lambda_{\min} = \ln \frac{a}{b}$. This yields the required bound on further simplification.
3. $\lambda_{\min} > 0$ for the lower tail bound as $c, d > 0$ and hence is within the $\lambda > 0$ range. However, for the upper tail bound, $\lambda_{\min} > 0$ and is within the permissible range only if $a > b$. We verify this for large n by substituting the values for a and b.

VIII. Concluding Remarks

In this work, we analysed content-wise hit-rates under the LRU policy based on the characteristic time of each content. We obtained an accurate approximation for the LRU content-wise hit-rates for large n, under Zipf distributed requests. To achieve this, we associated the problem of estimating the *characteristic time* of a content in the LRU policy with the classical Coupon Collector's Problem. Further, we provided analytical results in the form of tight concentration bounds on

ISBN 978-3-903176-37-9 © 2021 IFIP

the characteristic time about its mean to justify the accuracy of our approximations. Our bounds explicitly relate the accuracy of the proposed estimates to the cache-size scaling and the Zipf parameter β that governs the popularity distribution of the contents. In particular, we showed that the accuracy of the hit-rate estimate improves with a decrease in β or an increase in the cache-size scaling. The concentration bounds derived herein for the waiting time in the CCP under Zipf distributed coupon arrivals could be of independent interest, as the CCP is a classical problem with applications in several areas of engineering, for example, electrical fault detection, node discovery in wireless networks, etc., as well as in the fields of biology and linguistics.

A potential direction for future work is to move beyond the Zipf distribution, and to further consider requests that are correlated across time.

REFERENCES

[1] S. Melamed and Y. Bigio, "Bandwidth savings and qos improvement for www sites by catching static and dynamic content on a distributed network of caches," Jul. 16 2001, uS Patent App. 10/332,842.

[2] A. V. Aho, P. J. Denning, and J. D. Ullman, "Principles of optimal page replacement," *Journal of the ACM (JACM)*, vol. 18, no. 1, pp. 80–93, 1971.

[3] I. J. Good, "The population frequencies of species and the estimation of population parameters," *Biometrika*, vol. 40, no. 3-4, pp. 237–264, 1953.

[4] D. A. McAllester and R. E. Schapire, "On the convergence rate of good-turing estimators." in *COLT*, 2000, pp. 1–6.

[5] G. Valiant and P. Valiant, "Estimating the unseen: an n/log (n)-sample estimator for entropy and support size, shown optimal via new clts," in *Proceedings of the forty-third annual ACM symposium on Theory of computing*. ACM, 2011, pp. 685–694.

[6] G. Karakostas and D. Serpanos, "Practical lfu implementation for web caching," *Technical Report TR-622–00*, 2000.

[7] W. King, "Analysis of paging algorithms," *In Proc. IFIP Congress*, pp. 485–490, 1971.

[8] P. Flajolet, D. Gardy, and L. Thimonier, "Birthday paradox, coupon collectors, caching algorithms and self-organizing search," *Discrete Appl. Math.*, vol. 39, no. 3, pp. 207–229, Nov. 1992. [Online]. Available: http://dx.doi.org/10.1016/0166-218X(92)90177-C

[9] A. Dan and D. Towsley, "An approximate analysis of the lru and fifo buffer replacement schemes," *SIGMETRICS Perform. Eval. Rev.*, vol. 18, no. 1, pp. 143–152, Apr. 1990. [Online]. Available: http://doi.acm.org/10.1145/98460.98525

[10] R. Fagin, "Asymptotic miss ratios over independent references," *Journal of Computer and System Sciences*, vol. 14, no. 2, pp. 222–250, 1977.

[11] C. Berthet, "Contributions to the generalized coupon collector and lru problems," *arXiv preprint arXiv:1706.05250*, 2017.

[12] H. Che, Y. Tung, and Z. Wang, "Hierarchical web caching systems: modeling, design and experimental results," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 7, pp. 1305–1314, Sep 2002.

[13] M. Ferrante and M. Saltalamacchia, "The coupon collector's problem," *Materials matemàtics*, pp. 0001–35, 2014.

[14] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: evidence and implications," in *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 1, Mar 1999, pp. 126–134 vol.1.

[15] C. R. Cunha, A. Bestavros, and M. E. Crovella, "Characteristics of www client-based traces," Boston University Computer Science Department, Tech. Rep., 1995.

[16] D. N. Serpanos, G. Karakostas, and W. H. Wolf, "Effective caching of web objects using zipf's law," in *2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532)*, vol. 2, 2000, pp. 727–730 vol.2.

[17] C. Fricker, P. Robert, and J. Roberts, "A versatile and accurate approximation for lru cache performance," in *2012 24th International Teletraffic Congress (ITC 24)*, Sept 2012, pp. 1–8.

[18] B. Jiang, P. Nain, and D. Towsley, "Lru cache under stationary requests," *SIGMETRICS Perform. Eval. Rev.*, vol. 45, no. 2, p. 2426, Oct. 2017. [Online]. Available: https://doi.org/10.1145/3152042.3152051

[19] J. Tan, G. Quan, K. Ji, and N. Shroff, "On resource pooling and separation for lru caching," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 2, no. 1, Apr. 2018. [Online]. Available: https://doi.org/10.1145/3179408

[20] M. Brenner, "A lyapunov analysis of lru," Ph.D. dissertation, 2020.

[21] A. Wolman, M. Voelker, N. Sharma, N. Cardwell, A. Karlin, and H. M. Levy, "On the scale and performance of cooperative web proxy caching," in *Proceedings of the Seventeenth ACM Symposium on Operating Systems Principles*, ser. SOSP '99. New York, NY, USA: ACM, 1999, pp. 16–31. [Online]. Available: http://doi.acm.org/10.1145/319151.319153

[22] H. Gomaa, G. G. Messier, C. Williamson, and R. Davies, "Estimating instantaneous cache hit ratio using markov chain analysis," *IEEE/ACM Transactions on Networking*, vol. 21, no. 5, pp. 1472–1483, Oct 2013.

[23] V. S. Mookerjee and Y. Tan, "Analysis of a least recently used cache management policy for web browsers," *Oper. Res.*, vol. 50, no. 2, pp. 345–357, Mar. 2002. [Online]. Available: http://dx.doi.org/10.1287/opre.50.2.345.430

[24] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Transactions on Information Theory*, vol. 63, no. 11, pp. 7492–7508, 2017.

[25] R. D. Yates, P. Ciblat, A. Yener, and M. Wigger, "Age-optimal constrained cache updating," in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 141–145.

[26] P. Poojary, S. Moharir, and K. Jagannathan, "Caching policies under content freshness constraints," in *2018 10th International Conference on Communication Systems Networks (COMSNETS)*, Jan 2018, pp. 400–402.

[27] S. Fatale, S. Prakash, and S. Moharir, "Caching policies for transient data," in *2018 Twenty Fourth National Conference on Communications (NCC)*, Feb 2018, pp. 1–6.

[28] P. Poojary, S. Moharir, and K. Jagannathan, "Report with detailed proofs," 2021. [Online]. Available: https://drive.google.com/file/d/1SZthwMYpTNXpD381rzKrREYkncRRm4k1/view?usp=sharing

[29] S. Podlipnig and L. Böszörmenyi, "A survey of web cache replacement strategies," *ACM Comput. Surv.*, vol. 35, no. 4, pp. 374–398, Dec. 2003. [Online]. Available: http://doi.acm.org/10.1145/954339.954341

[30] L. A. Adamic and B. A. Huberman, "Zipf's law and the internet." *Glottometrics*, vol. 3, no. 1, pp. 143–150, 2002.

[31] P. Erdős and A. Rényi, "On a classical problem of probability theory," 1961.

[32] L. E. Baum and P. Billingsley, "Asymptotic distributions for the coupon collector's problem," *The Annals of Mathematical Statistics*, vol. 36, no. 6, pp. 1835–1839, 1965.

[33] A. Pósfai, "Approximation theorems related to the coupon collector's problem," *arXiv preprint arXiv:1006.3531*, 2010.

[34] A. Boneh and M. Hofri, "The coupon-collector problem revisiteda survey of engineering problems and computational methods," *Stochastic Models*, vol. 13, no. 1, pp. 39–66, 1997.

ISBN 978-3-903176-37-9 © 2021 IFIP