Asymptotic Miss Ratio of LRU Caching with Consistent Hashing

Kaiyi Ji, Guocong Quan, Jian Tan
Department of Electrical and Computer Engineering
The Ohio State University, Columbus, 43210
Email: {ji.367, quan.72, tan.252}@osu.edu

***Abstract*—To efficiently scale data caching infrastructure to support emerging big data applications, many caching systems rely on consistent hashing to group a large number of servers to form a cooperative cluster. These servers are organized together according to a random hash function. They jointly provide a unified but distributed hash table to serve swift and voluminous data item requests. Different from the single least-recently-used (LRU) server that has already been extensively studied, theoretically characterizing a cluster that consists of multiple LRU servers remains yet to be explored. These servers are not simply added together; the random hashing complicates the behavior. To this end, we derive the asymptotic miss ratio of data item requests on a LRU cluster with consistent hashing. We show that these individual cache spaces on different servers can be effectively viewed as if they could be pooled together to form a single virtual LRU cache space parametrized by an appropriate cache size. This equivalence can be established rigorously under the condition that the cache sizes of the individual servers are large. For typical data caching systems this condition is common. Our theoretical framework provides a convenient abstraction that can directly apply the results from the simpler single LRU cache to the more complex LRU cluster with consistent hashing.**

I. Introduction

With the advent of cloud computing and emergence of big data, scale-out data caching systems are widely deployed and their horizontal scalability [1] becomes increasingly important. As an effective solution, consistent hashing [2] has been commonly used by key-value caching systems, e.g., Dynamo [3], Aerospike [4], Memcached [5], Redis [6]. Using consistent hashing, a large number of servers are organized together to form a cooperative cluster. These servers jointly provide a unified but distributed hash table to serve swift and voluminous data item requests. Once a data item is hashed to one of the hosting servers, most key-value caching systems use the least-recently-used (LRU) caching algorithm, or its variations, e.g., LRU Clock [7], to decide which data items should be kept in its own individual cache space. These data caching systems play a critical role in optimizing the way information is delivered in Web services.

With consistent hashing, the total amount of cache spaces in the cluster can be easily expanded (scaled horizontally) through the addition of new cache servers. However, these individual cache spaces on different servers are not simply added together to achieve an overall request miss ratio. Although LRU caching on a single server has already been extensively studied, theoretically characterizing the miss ratio of a LRU cluster organized by consistent hashing still remains an unexplored problem. One difficulty in analysis is that the data item requests are shuffled to a large number of servers according to a random hash function. This random hashing complicates the system behavior. Due to the fundamental role and predominant usage in practice, LRU caching with consistent hashing merits a deep investigation.

Characterizing the cache miss behavior of a cluster in such a complex setting not only helps resource planning but also improves the way a cache cluster is organized. To this end, we derive the asymptotic miss ratio of a LRU caching cluster with consistent hashing under the independent reference model (IRM) [8]. Interestingly, these individual cache spaces on different servers, though isolated physically but logically connected through a hash function, can be effectively viewed as if they could be pooled together to form a single virtual LRU cache space. Interestingly, this virtual LRU cache has a equivalent cache size determined by the distribution of the random hash function. This equivalence can be established when the cache sizes of the individual servers are large. Our result provides a convenient abstraction that can rigorously relate the more complex LRU caching with consistent hashing to the relatively simpler single LRU cache. Based on this abstraction, many known results on a single LRU cache can be directly translated to a LRU cluster with consistent hashing. Specifically, we prove a characteristic time approximation, previously established for a single LRU cache, for a LRU cluster. Notably, the characteristic time approximation has the same form for almost all of the random hash functions. This result is not straightforward in view that the miss ratio of each of the server is a random variable conditional on the random hash function. However, the overall conditional asymptotic miss ratio of the cluster is always the same almost surely, depending on the probability distribution of the random hash function. Due to this equivalence, we comment that the engineering implications discussed in [9] for a single LRU cache can also be extended to a LRU cluster.

A. Background

To put the analysis on a concrete basis, we first summarize the important features of consistent hashing and LRU caching.

This work was supported by the National Science Foundation under Grant No. 1717060.

978-1-5386-4128-6/18/$31.00 ©2018 IEEE

1) Consistent hashing: Data items are usually organized in a key value pair, and the entire data are stored in the whole cluster as a distributed hash table according to keys. To

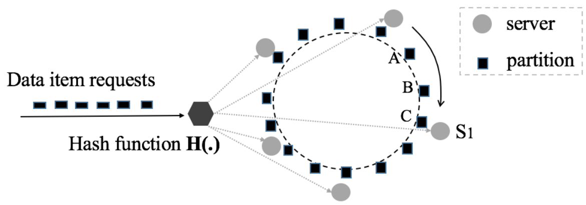

Fig. 1: Illustration of consistent hashing

scale out the system horizontally, each server is maintained independent, e.g., by using consistent hashing [2] to select a unique server for each key. The basic idea is to first hash the data items to a large number of partitions at random. These partitions form a ring, as illustrated in Fig. 1. The physical servers, with a total number that is much smaller than the number of partitions, are also hashed to a subset of the partitions. A data item, after being hashed to a partition, will be stored on the server that is closest in the clockwise direction to its associated partition. For example, partitions A, B and C are all stored on server S_1 in Fig.1. Therefore, when the server locations are fixed, each server hosts a certain number of partitions that are determined by the hash function.

2) LRU caching: Each data item is hashed to one of the hosting servers. A cache replacement algorithm is needed to manage the cache space on an individual server. Due to the low cost of tracking access history, LRU caching algorithm has been widely used, e.g., for Memcached [5], [10]. A data item is added to the cache after a client has requested it and failed. When the cache is full, the LRU caching algorithm evicts a data item that has not been used for the longest time in order to accommodate the newly requested one.

B. Contributions of this paper

(I) For a family $\mathcal{H}$ of hash functions under the Simple Uniform Hashing Assumption (SUHA) [11], we characterize the asymptotic miss ratio of data item requests on a cluster with consistent hashing. This asymptotic result, expressed as a conditional probability, holds almost surely for all random hash functions in $\mathcal{H}$. It provides a new analytical framework to study LRU caching with consistent hashing by conditioning on the random hash function, which is an interesting feature that most existing asymptotic results do not have.

(II) We rigorously establish a one-to-one equivalence between a cluster with consistent hashing and a single virtual LRU cache space with a proper size parametrized by the distribution of the random hash function. Conveniently, this equivalence translates the analytic results from the well-studied single server to the complex cluster with consistent hashing. Based on this equivalence, we prove the characteristic time approximation for a cluster to characterize the miss probability.

(III) Extensive simulations show that our asymptotic results match with the empirical results accurately even for relatively small cache sizes.

C. Related work

Consistent hashing has gained much popularity in recent years due to the increasing demand of processing large data sets on a scale-out infrastructure. It has been successfully used in a number of real-world applications, e.g., web caches [12], [13], peer-to-peer networks [14] and distributed storage systems [15]. Most theoretical studies on consistent hashing focus on characterizing the randomized partitioning algorithms to balance data allocation [16], memory sharing [17] and perfect hashing [18]. These algorithms are usually analyzed under SUHA. Some works circumvent this assumption using realistic hash functions through simulations [19], [20]. However, none of these works investigate the miss probabilities of a cluster with consistent hashing.

There is a large body of work on the miss ratio of a single LRU server. Different methods have been proposed, e.g., approximation by iterative algorithms [21], mean field analysis [22] and the characteristic time approximation [23], [24]. To obtain insights, asymptotic results for Zipf's popularity distributions have been derived [25], [26], [27]. The characteristic time approximation is also a common approach, which has been shown to be accurate in practical applications [28], [29]. Its success has been supported by the analysis [30], [31]. Nevertheless, these results cannot be directly extended to a cluster with consistent hashing.

Although single LRU caches have been studied in depth, characterizing the miss behavior of cache networks with general topologies remains difficult. Instead, some existing works, e.g., [32], [33], focus on offline optimization problems (e.g., content placement) in cache networks. Some specially structured cache networks (e.g., tree or line networks) have been studied [34], [35] using a TTL-based eviction scheme. With consistent hashing, the cache network can be viewed as a one-hop network, which is the focus of this paper.

II. Model description

Consider a cluster $\mathcal{C}$ with N servers $\{S_1, S_2, \cdots, S_N\}$ organized by consistent hashing. Assume that the data item requests hosted on $\mathcal{C}$ can access an infinite number of distinct data items of unit size that are represented by a sequence $(d_i^\circ, i = 1, 2, 3, \cdots)$. The notation $H(d_i^\circ) = m$ represents that a data item d_i°, together with all requests that ask for d_i°, are hashed to server m by the hash function $H(\cdot)$. Thus, a subsequence of data items, denoted by $(d_i^{(m)}, i = 1, 2, \cdots)$, are hashed to server S_m, selected from $(d_i^\circ, i = 1, 2, \cdots)$.

To characterize the hash function H, we assume the Simple Uniform Hashing Assumption (SUHA) to facilitate the analysis. In practice, the ideal SUHA property is not feasible, and people resort to a (strongly) universal hash family [36] or a k-independent hash family [37] as approximations. Actually, it has been shown that 2-independent hash functions under mild conditions approximate truly random functions [38]. Specifically, we consider a family of hash functions $\mathcal{H} = \{h_w(\cdot), w = 1, 2, \cdots\}$. Assume that, with H chosen from $\mathcal{H}$ uniformly at random, each data item $d_i^\circ, i = 1, 2, \cdots$ is dispatched also uniformly at random to one of the partitions.

Note that although $H(\cdot)$ is random, it becomes one of the deterministic hash function $h_w(\cdot)$ once the random selection is completed. Using consistent hashing, the involved servers store the data items from mutually exclusive subsets of the partitions, as shown in Fig. 1. Since the number of partitions assigned to each of the servers could be different, we can equivalently assume that each data item d_i° is independently hashed to server $S_m, 1 \le m \le N$ with probability μ_m by $H(\cdot)$. In other words, we have the following assumption.

Assumption 1. *(SUHA) $H(d_i^\circ), i = 1, 2, \cdots$ are independent random variables with $\mathbb{P}[H(d_i^\circ) = m] = \mu_m, 1 \le m \le N$.*

Assume that the arrivals of the data item requests occur at time points $\{\tau_n, -\infty < n < +\infty\}$. Let J_n be the index of the server for the request at time τ_n. The event $\{J_n = m\}$ represents that the request at time τ_n is processed on server S_m. Denote by R_n the requested data item at time τ_n. Thus, the event $\{J_n = m, R_n = d_i^{(m)}\}$ means that the request at time τ_n is to fetch data item $d_i^{(m)}$ on server S_m. In order to compute the miss ratio when the system reaches stationarity, we consider the request at time τ_0. It has been shown [26] that the miss ratio is equal to the probability that the data item requested by R_0 is not in the cache. For cluster $\mathcal{C}$, define

$$\mathbb{P}\left[R_0 = d_i^{(m)} \big| J_0 = m, H\right] = q_i^{(m)}, i = 1, 2, 3, \cdots \quad (1)$$

$$\mathbb{P}\left[R_0 = d_i^\circ\right] = q_i^\circ, i = 1, 2, 3, \cdots \quad (2)$$

Note that $\left(q_i^{(m)}, i \ge 1\right)$ is a random sequence determined by the random hash function H. We assume that the data items $(d_i^\circ, i \ge 1)$ are sorted such that the sequence $(q_i^\circ, i \ge 1)$ is non-increasing with respect to i. Since $\left(d_i^{(m)}, i \ge 1\right)$ is a random subsequence of $(d_i^\circ, i \ge 1)$, $\left(q_i^{(m)}, i \ge 1\right)$ is also non-increasing by this ordering. Let $\left(d_{m_i}^\circ, i \ge 1\right) \equiv \left(d_i^{(m)}, i \ge 1\right)$, which represents the subsequence of $(d_i^\circ, i \ge 1)$ that is hashed to server S_m by H. Therefore, $\mathbb{P}[J_0 = m \big| H] = \sum_{i=1}^{\infty} q_{m_i}^\circ$. For notational convenience in our proofs, we define $W_m = 1/\mathbb{P}[J_0 = m \big| H]$. On server S_m, we have $q_i^{(m)} = W_m q_{m_i}^\circ$, $i = 1, 2 \cdots$. We emphasize that W_m is a random variable determined by the random hash function H, which normalizes $Q_m = \left(q_{m_i}^\circ, i \ge 1\right)$ to be a legitimate distribution.

The data item popularity is assumed to follow a Zipf's distribution $q_i^\circ \sim c_\circ / i^{\alpha_\circ}$. This is a typical distribution that has been empirically observed in web pages [39], content-centric network [40], and video systems [41]. To simplify the analysis, this paper only considers a Zipf's distribution with $\alpha_\circ > 1$. For $\alpha_\circ < 1$, we can conduct a similar analysis based on existing results [42], [43], [44].

LRU is equivalent to the move-to-front (MTF) policy [26], [45], which sorts the data items in an increasing order of their last access time. When a data item is requested under MTF, it is moved to the first position of the list and all the other data items that were in front of this one increase their positions by one. Define C_n to be the position of the data item requested by R_n in the sorted list under MTF on the server that processes the request R_n. Then, the miss probability of the requests on server S_m with a cache size x_m is given by $\mathbb{P}[C_0 > x_m \big| J_0 = m, H]$, which is conditional on the random hash function H and the event that R_0 occurs on server S_m, i.e., $J_0 = m$. Combining the miss ratios of the servers, we obtain the overall miss probability of the cluster $\mathcal{C}$, conditional on $H \in \mathcal{H}$,

$$\mathbb{P}_{miss}^{\mathcal{C},H} = \sum_{m=1}^{N} \mathbb{P}\left[C_0 > x_m \big| J_0 = m, H\right] \mathbb{P}\left[J_0 = m \big| H\right]. \quad (3)$$

In the analysis, we assume $x_m = b_m x, 1 \le m \le N$ for $x > 0$.

III. Main results

In this section, we first derive the miss ratio for each of the servers of the LRU cluster with consistent hashing conditional on the random hash function. Then, we show that these individual cache servers can be regarded as a single virtual LRU server with a proper cache size. This connection also proves the characteristic time approximation for a cluster.

A. Asymptotic miss ratio under random hashing

We derive the miss probabilities for the servers of the cluster by conditioning on the random hash function H. Note that H uniquely determines $W_m, 1 \le m \le N$. The gamma function is given by $\Gamma(\alpha + 1) = \int_0^\infty y^\alpha e^{-y} dy$. The notation $f(x) \sim g(x)$ means $\lim_{x \to \infty} f(x)/g(x) = 1$.

Theorem 1. *Under the assumptions in Section II, we obtain, for all $1 \le m \le N$, almost surely for all H, as $x_m \to \infty$,*

$$\mathbb{P}\left[C_0 > x_m \big| J_0 = m, H\right] \sim \frac{(\mu_m \Gamma(1 - 1/\alpha_\circ))^{\alpha_\circ} c_\circ W_m}{\alpha_\circ x_m^{\alpha_\circ - 1}}, \quad (4)$$

which implies, almost surely for all H,

$$\mathbb{P}_{miss}^{\mathcal{C},H} \sim \sum_{m=1}^{N} \frac{\mu_m^{\alpha_\circ} \Gamma(1 - 1/\alpha_\circ)^{\alpha_\circ} c_\circ}{\alpha_\circ x_m^{\alpha_\circ - 1}}. \quad (5)$$

Proof. The proof is presented in Section VI-A. □

This asymptotic result in (4) involves random variables $W_m, 1 \le m \le N$ that are determined by H. Interestingly, the overall asymptotic miss ratio of the whole cluster in (5) is independent of H since $W_m = 1/\mathbb{P}[J_0 = m \big| H]$. These asymptotic results hold a.s. for all H. See Experiments in Section IV. If there is only one server in the cluster $\mathcal{C}$, i.e., $N = 1$, Theorem 1 reproduces the results in [25], [26] for a Zipf's distribution, e.g., Theorem 3 of [26] on an asymptotic miss probability of a single LRU server. However, extending this result from a single server to a cluster is complicated. We discuss two main issues that cause the difficulty: 1) Theorem 3 of [26] assumes a deterministic popularity distribution on a server. This condition is not satisfied in our model due to the random hash function; 2) the proofs of [25], [26] cannot be used to prove the characteristic time approximation for a cluster. Because of these reasons, we use a different approach to derive the miss probability of a LRU cluster with consistent hashing, which also proves the characteristic time approximation for a cluster.

Now, suppose that we have a single virtual LRU cache server of size $\bar{x}$ that serves the entire data item requests $\{R_n\}$, which at the same time are also served on the cluster $\mathcal{C}$. Based on Theorem 1, we establish an equivalence between the cluster $\mathcal{C}$ and the virtual LRU cache. Denote by $\mathbb{P}\left[C_0 > \bar{x}\ \right]$ the miss probability of the virtual LRU cache conditional on H. Recall that the server S_m has a cache capacity $x_m = b_m x$.

Theorem 2. *Under the assumptions in Section II, we obtain, almost surely for all H,*

$$\mathbb{P}_{miss}^{\mathcal{C},H} \sim \mathbb{P}\left[C_0 > \bar{x}\right], \ \textit{as}\ x \to \infty, \tag{6}$$

where

$$\bar{x} = x\Big(\sum_{m=1}^{N} \mu_m^{\alpha_\circ} b_m^{1-\alpha_\circ}\Big)^{-1/(\alpha_\circ - 1)}. \tag{7}$$

Proof. The proof is presented in Section VI-B. □

This theorem shows that the miss probability on the cluster $\mathcal{C}$ is asymptotically equal to the miss ratio of a LRU server with the cache size given by (7). Interestingly, as illustrated in Experiment 2, this asymptotic equivalence is accurate even when the cache size of each individual server of cluster $\mathcal{C}$ is relatively small. Using this connection, existing results and insights that have been established for a single server seem to be also true for a LRU cluster with consistent hashing. This could be useful for resource planning and cluster optimization.

B. Characteristic time approximation with consistent hashing

The characteristic time approximation [24] has been widely used in estimating the miss ratio of a LRU server. Based on Theorem 2, we derive the characteristic time approximation for a cluster.

Theorem 1 shows that, although the miss ratio of each server is random, determined by H, the overall asymptotic miss ratio of the cluster is independent of H. This interesting result motivates us to define the characteristic time approximation for the cluster $\mathcal{C}$ as

$$\mathbb{P}_{\text{CT}}\left[C_0 > \bar{x}\right] = \sum_{i=1}^{\infty} q_i^\circ e^{-q_i^\circ t_C}, \tag{8}$$

where $\bar{x}$ is given by (7) and t_C is the unique solution of the equation $\sum_{i=1}^{\infty}(1 - e^{-q_i^\circ t_C}) = \bar{x}$.

Theorem 3. *Under the assumptions of Theorem 2, we have, almost surely for all H,*

$$\mathbb{P}_{CT}\left[C_0 > \bar{x}\right] \sim \mathbb{P}_{miss}^{\mathcal{C},H}, \ \textit{as}\ \bar{x} \to \infty. \tag{9}$$

Proof. The proof is presented in Section VI-C. □

IV. Simulations

In this section, we conduct extensive simulations using C++ to verify the main results in Section III. Notably, all simulations match with our theoretical results even for relatively small cache sizes.

Experiment 1. This experiment verifies Theorem 1. Consider a cluster of 100 heterogeneous servers $\{S_1, S_2, \cdots, S_{100}\}$ that have distinct cache sizes and different hashing probabilities. The server S_m, $1 \le m \le 100$ has a cache capacity $x_m = (1 + 0.1 z_m)\, x$ with z_m selected uniformly at random from $[-0.5, 0.5]$. Thus, x is the average cache size across all of the servers. Recall that μ_m is the probability that a data item is hashed to server S_m. Let $\mu_m =$

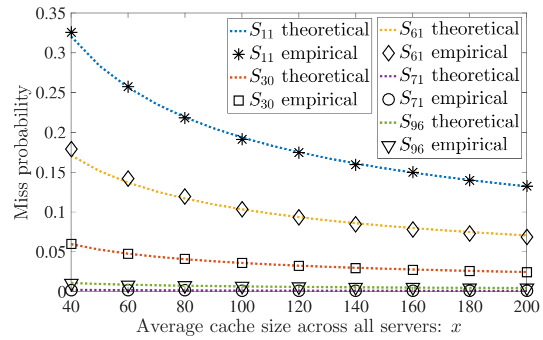

Fig. 2: Miss probabilities of five selected servers

$(0.1 + 0.05\lfloor(m-1)/20\rfloor)/20$ for $1 \le m \le 100$. Conditional on H, we obtain random variables $W_m, 1 \le m \le 100$ with $W_m = 1/\sum_{i=1}^{\infty} q_{m_i}^\circ$. Set the total number of data items $M = 10^7$, and the popularity distribution $q_i^\circ = c_\circ/i^{\alpha_\circ}, 1 \le i \le M$ with $\alpha_\circ = 1.55, c_\circ = 1/(\sum_{i=1}^{M} i^{-\alpha_\circ}) = 0.4109$. For each $x \in \{40, 60, 80, 100, 120, 140, 160, 180, 200\}$, we first simulate 10^8 requests to ensure that the entire system reaches stationary, and then 10^9 more requests to compute the empirical miss probabilities of the cluster $\mathcal{C}$ and the individual servers. To verify (4), we need to show that it holds for all $1 \le m \le 100$. To visualize the results, we only plot the miss probabilities of five servers $\{S_{11}, S_{30}, S_{61}, S_{71}, S_{96}\}$ in Fig. 2. The empirical results match well with the the theoretical results by (4) and (5) even when x is small.

Experiment 2. This experiment verifies the equivalence between the cluster $\mathcal{C}$ and a virtual LRU cache described in Theorem 2, by using the same setting as in Experiment 1. We

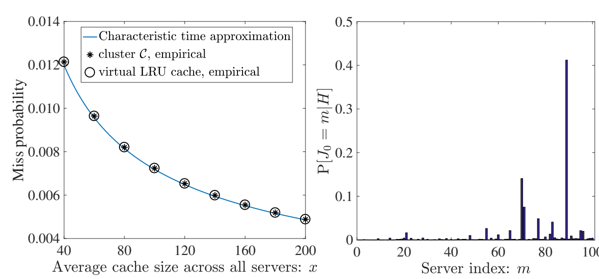

Fig. 3: [Left] Miss ratios of the cluster $\mathcal{C}$, a virtual LRU cache and the characteristic time approximation. [Right] The probability to hash a request to server S_m conditional on H

demonstrate the accuracy of the characteristic time approximationin, which also verifies Theorem 3. By computing (7), we obtain the equivalent size of the virtual LRU caching space $\bar{x} = 91.1784x$. For the characteristic time approximation of the cluster $\mathcal{C}$, we use a binary search to find the solution t_C of the equation $\sum_{i=1}^{\infty}(1-e^{-q_i^\circ t_C}) = 91.1784x$ and then calculate the miss probability by (8). It can be shown from the left figure in Fig. 3 that the empirical results match well with the theoretical results for the miss probabilities of the virtual LRU caching server and the cluster $\mathcal{C}$ even for $x = 40$. In addition, the characteristic time approximation (8) provides an accurate estimation of the miss ratio of the cluster $\mathcal{C}$.

Experiment 3. This experiment moves beyond the assumptions of this paper and considers a realistic setting. Thus, we cannot explicitly compute the equivalent virtual cache size by Theorem 2. However, we still demonstrate an equivalence between the cluster $\mathcal{C}$ and a virtual cache. We set $\alpha_\circ = 0.8$ and use a 2-independent hash function [38]. For a cluster of 100 servers $\{S_1, S_2, \cdots, S_{100}\}$ described in Experiment 1, we hash each server to one of 2000 partitions using the 2-independent hash function $h_{a,b}(S_i) = N_i = ((a \times i + b) \mod p) \mod 2000$, where a, b are chosen from $\{1, 2, \cdots, p\}$ uniformly at random with a large prime $p = 15881$. Using the same hash function, each data item d_i° is hashed to one of these partitions. The data items from partition k are stored on the server that has an index $\arg\min_i\{N_i : N_i \geq k\}$ if the set $\{i : N_i \geq k\} \neq \emptyset$ and $\arg\min_i\{N_i\}$ otherwise. We set the total number of data items $M = 10^4$, and the

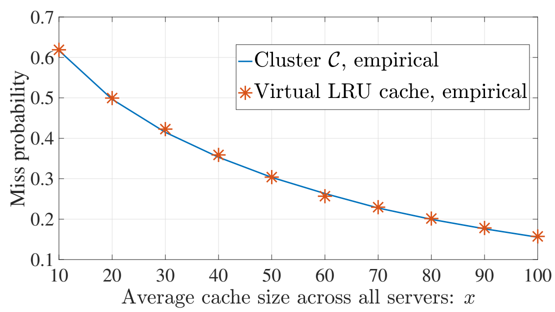

Fig. 4: Miss ratios of the cluster $\mathcal{C}$ and a virtual LRU cache

popularity distribution $q_i^\circ = c_\circ/i^{0.8}, 1 \leq i \leq M$ with $c_\circ = 1/(\sum_{i=1}^{M} i^{-0.8}) = 0.0369$. For the virtual LRU caching space, we find the equivalent size $\bar{x} = 70.0212x$. It can be shown from Fig. 4 that the empirical miss ratios of the cluster $\mathcal{C}$ and the virtual LRU cache match very well.

V. Conclusion

Driven by the trend to scale out caching systems for processing big data, LRU caching with consistent hashing has been widely deployed. We develop a theoretical framework to investigate the miss ratio of a LRU cluster for a family of hash functions satisfying the Simple Uniform Hashing Assumption (SUHA). We derive a close-form asymptotic miss probability that holds almost surely for all of the random hash functions from this family. This result also establishes a one-to-one equivalence between a LRU cluster and a single virtual LRU server. It provides a convenient abstraction to understand the complex LRU cluster using the insights obtained from a LRU server. Based on this connection, we also prove the characteristic time approximation for a cluster with consistent hashing.

VI. Proofs

This section contains the proofs of our main theorems.

A. Proof of Theorem 1

We rely on the following Lemma 1 to prove Theorem 1. To this end, we use Lemmas 5, 6 and 7 to study the three quantities in Lemma 1, i.e., q_i/q_{i+1}, $\Phi(\cdot)$ and $T(x)$. Specifically, we first note that the variables $\left(I_k^{(m)}, k \geq 1\right)$ can be regarded as the indices of data items on server S_m and show that $1 \leq q^{(m)}_{I_k^{(m)}}/q^{(m)}_{I_k^{(m)}+1} \leq 1+\epsilon$ holds with high probability $1 - c_1/n^2$ in Lemma 5. Then, we prove that the functional relationship $\tilde{\Phi}_m(z)$ between $\left(\sum_{i=I_k^{(m)}+1}^{\infty} q_i^{(m)}\right)^{-1}$ and $\left(q^{(m)}_{I_k^{(m)}}\right)^{-1}$ satisfies $\tilde{\Phi}_m(z) \sim az^\beta$ with high probability $1 - c_2/n^2$ in Lemma 6. Last, we show $T_m(z) \sim bz^\alpha$ with high probability $1 - c_3/n^2$ in Lemma 7. Using $\sum_{n=1}^{\infty}(c_1+c_2+c_3)/n^2 < \infty$ and Borel-Cantelli lemma, we prove that Theorem 1 holds almost surely for all $H \in \mathcal{H}$.

Lemma 1 is a direct consequence of Theorem 1 in [9]. Let the data item popularity distribution on a server be $(q_i, i \geq 1)$. Consider the following functional relationship

$$\left(\sum_{i=y}^{\infty} q_i\right)^{-1} \sim \Phi\left(q_y^{-1}\right), \ y \to \infty. \quad (10)$$

Define an increasing function $T(x) = \sum_{i=1}^{\infty}\left(1 - (1-q_i)^x\right)$ with an inverse $T^{\leftarrow}(x)$. We say that $f(x) \lesssim g(x)$ as $x \to \infty$ if $\limsup_{x\to\infty} f(x)/g(x) \leq 1$; $f(x) \gtrsim g(x)$ has a complementary definition.

Lemma 1. *If* $1 \leq \lim_{i\to\infty} q_i/q_{i+1} < 1+\epsilon$, $T(z) \sim c_1 z^\alpha, \alpha > 0$ *and* $\Phi(z) \sim c_2 z^\beta, \beta > 0$, *then, as* $x \to \infty$,

$$\frac{\Gamma(1+\beta)(1+\epsilon)^{-1}}{\Phi(T^{\leftarrow}(x))} \lesssim \mathbb{P}[C_0 > x] \lesssim \frac{\Gamma(1+\beta)(1+\epsilon)}{\Phi(T^{\leftarrow}(x))}. \quad (11)$$

Proof. The proof is based on Theorem 1 in [9]; see the technical report [46] for the details. □

We introduce some necessary definitions. Define mutually independent Bernoulli random variables $\{X_i^{(m)}\}$. Let $X_i^{(m)} = 1$ indicate that the data item d_i° is hashed to server S_m and $X_i^{(m)} = 0$ otherwise. We have $\mathbb{P}[X_i^{(m)} = 1] = \mu_m$. Define $I_k^{(m)} \triangleq \sum_{i=1}^{k} X_i^{(m)}$, which represents the number of data items hashed to server S_m from $(d_i^\circ, 1 \leq i \leq k)$. Let $Z_i^{(m)} \triangleq q_i^\circ X_i^{(m)}$ and $Y_n^{(m)} \triangleq \sum_{i=n}^{\infty} Z_i^{(m)}$. We quote the Bernstein's inequality in Lemma 2, and establish the following Lemma 3 to estimate $Y_k^{(m)}$, which will be used to estimate $\sum_{i=I_k^{(m)}+1}^{\infty} q_i^{(m)}$ in (29).

Lemma 2 (Theorem 2.8 in [47])**.** *For independent random variables* $X_i \leq M, 1 \leq i \leq n$ *with* $X = \sum_{i=1}^{n} X_i$, *we obtain,* $\forall\, \epsilon > 0$,

$$\mathbb{P}\left[X - \mathbb{E}[X] > \epsilon\right] \leq \exp\left(-\frac{\epsilon^2}{2\sum_{i=1}^{n}\mathbb{E}[X_i^2] + \frac{2M\epsilon}{3}}\right). \quad (12)$$

Lemma 3. *There exist $n_0 \in \mathbb{N}^+$ and $c_0 > 0$ such that for $\forall\, n > n_0$,*

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{\left|Y_k^{(m)} - \mathbb{E}[Y_k^{(m)}]\right| < 1/k^{\alpha_\circ - \frac{2}{3}}\right\}\right] > 1 - \frac{c_0}{n^2}. \quad (13)$$

Proof. Define events $\mathcal{A}_k \triangleq \{|Y_k^{(m)} - \mathbb{E}[Y_k^{(m)}]| < 1/k^{\alpha_\circ - \frac{2}{3}}\}$, $k \geq 1$. Recalling q_i° is non-increasing with respect to i, we have, for all $i \geq n$

$$Z_i^{(m)} \leq q_i^\circ \leq q_n^\circ \lesssim c_\circ/n^{\alpha_\circ}. \quad (14)$$

Noting that $\mathbb{E}[(Z_i^{(m)})^2] = \mu_m(q_i^\circ)^2 \sim \mu_m c_\circ^2/i^{2\alpha_\circ}$, we have

$$\sum_{i=n}^{\infty}\mathbb{E}[(Z_i^{(m)})^2] \sim \int_n^\infty \frac{\mu_m c_\circ^2}{x^{2\alpha_\circ}}dx = \frac{\mu_m c_\circ^2}{(2\alpha_\circ - 1)n^{2\alpha_\circ - 1}}. \quad (15)$$

Applying (12) for the random variable $Y_n^{(m)} = \sum_{i=n}^\infty Z_i^{(m)}$ and using (14) and (15), we have, there exist $n_{i_1} \in \mathbb{N}^+$ and $v > 0$ such that for $\forall\, n > n_{i_1}$,

$$\mathbb{P}\left[Y_n^{(m)} - \mathbb{E}[Y_n^{(m)}] > \frac{1}{n^{\alpha_\circ - 2/3}}\right] < \exp(-n^{1/3}v). \quad (16)$$

In the meanwhile, applying (12) for $-Y_n^{(m)} = \sum_{i=n}^\infty(-Z_i^{(m)})$ and using $-Z_i^{(m)} < 0$ and (15), we have, for $\forall\, n > n_{i_1}$,

$$\mathbb{P}\left[Y_n^{(m)} - \mathbb{E}[Y_n^{(m)}] < -\frac{1}{n^{\alpha_\circ - 2/3}}\right] < \exp(-n^{1/3}v), \quad (17)$$

which, in conjunction with (16), implies, for $n > n_{i_1}$

$$\mathbb{P}\left[\left|Y_n^{(m)} - \mathbb{E}[Y_n^{(m)}]\right| \geq \frac{1}{n^{\alpha_\circ - 2/3}}\right] < 2\exp(-n^{\frac{1}{3}}v). \quad (18)$$

Using (18) and a union bound, we obtain, for $\forall n > n_{i_1}$

$$\begin{aligned}\mathbb{P}\left[\{\cap_{k\geq n}\mathcal{A}_k\}^c\right] &= \mathbb{P}\left[\cup_{k\geq n}\left\{\left|Y_k^{(m)} - \mathbb{E}[Y_k^{(m)}]\right| \geq 1/k^{\alpha_\circ - 2/3}\right\}\right]\\ &\leq \sum_{k=n}^\infty \mathbb{P}\left[\left|Y_k^{(m)} - \mathbb{E}[Y_k^{(m)}]\right| \geq 1/k^{\alpha_\circ - 2/3}\right]\\ &< \sum_{k=n}^\infty 2e^{-k^{1/3}v} < \int_n^\infty \exp(-x^{1/3}v)dx. \quad (19)\end{aligned}$$

There exist n_{i_2} and $c_0 > 0$ such that $\int_n^\infty \exp(-x^{1/3}v)dx < c_0/n^2$ holds for $n > n_{i_2}$. Using (19) and letting $n_0 \triangleq \max\{n_{i_1}, n_{i_2}\}$, we finish the proof. ☐

We establish the following lemma 4 to estimate $q^{(m)}_{I_k^{(m)}}$, which is used to estimate $\tilde{\Phi}_m(z)$ and the ratio $q^{(m)}_{I_k^{(m)}}/q^{(m)}_{I_k^{(m)}+1}$.

Lemma 4. *There exist $n_1 \in \mathbb{N}^+$ and $c_1 > 0$ such that for $\forall\, n > n_1$,*

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{W_m q_k^\circ \leq q^{(m)}_{I_k^{(m)}} < W_m q^\circ_{k-\lceil k^{1/2}\rceil+1}\right\}\right] < 1 - \frac{c_1}{n^2}. \quad (20)$$

Proof. Define events $\mathcal{B}_k \triangleq \{q^{(m)}_{I_k^{(m)}} \geq W_m q^\circ_{k-\lceil k^{1/2}\rceil+1}\}$. Let $\mathcal{C}_k \triangleq \left\{H(d_i^\circ) \neq S_m, k - \lceil k^{1/2}\rceil + 1 \leq i \leq k\right\}$ be the event that none of data items $d_i^\circ, k - \lceil k^{1/2}\rceil + 1 \leq i \leq k$ are hashed to S_m. Next, we will show that $\mathcal{B}_k$ and $\mathcal{C}_k$ are equivalent.

Since both $q_i^{(m)}$ and q_i° are non-increasing with respect to i, the event $\mathcal{B}_k$ implies,

$$d_i^{(m)} \notin \left\{d_j^\circ, k - \lceil k^{1/2}\rceil + 1 \leq j \leq k\right\}, 1 \leq i \leq I_k^{(m)}. \quad (21)$$

Moreover, based on the definition of $I_k^{(m)}$, we have, $\{d_i^{(m)}, i \geq I_k^{(m)} + 1\} \subseteq \{d_i^\circ,\, i \geq k+1\}$, which implies,

$$d_i^{(m)} \notin \left\{d_j^\circ, k - \lceil k^{1/2}\rceil + 1 \leq j \leq k\right\}, i \geq I_k^{(m)} + 1. \quad (22)$$

Combining (21) and (22) yields $\mathcal{B}_k \subseteq \mathcal{C}_k$. On the other side, the event $\mathcal{C}_k$ implies $q_i^{(m)} \geq W_m q^\circ_{k-\lceil k^{1/2}\rceil+1}$ for all $1 \leq i \leq I_k^{(m)}$, yieding $\mathcal{C}_k \subseteq \mathcal{B}_k$. Thus, we have $\mathcal{B}_k$ is equivalent to $\mathcal{C}_k$. Under Assumption 1, we have $\mathbb{P}[\mathcal{B}_k] = \mathbb{P}[\mathcal{C}_k] = (1-\mu_m)^{\lceil k^{1/2}\rceil}$. Noting the complement $\mathcal{B}_k^c = \{W_m q^\circ_{k-\lceil k^{1/2}\rceil+1} > q^{(m)}_{I_k^{(m)}} \geq W_m q_k^\circ\}$ and using a union bound, we obtain,

$$\begin{aligned}\mathbb{P}\left[\cap_{k\geq n}\mathcal{B}_k^c\right] &= 1 - \mathbb{P}\left[\cup_{k\geq n}\mathcal{B}_k\right] \geq 1 - \sum_{k\geq n}(1-\mu_m)^{\lceil k^{1/2}\rceil}\\ &\geq 1 - \sum_{k\geq n}(1-\mu_m)^{k^{1/2}} \geq 1 - \int_{n-1}^\infty (1-\mu_m)^{x^{1/2}}dx \quad (23)\end{aligned}$$

There exist a large integer n_1 and a constant $c_1 > 0$ such that for all $n \geq n_1$, $\int_{n-1}^\infty (1-\mu_m)^{x^{1/2}}dx < c_1/n^2$, which, in conjunction with (23), completes the proof. ☐

We use the following Lemma 5 to estimate $q^{(m)}_{I_k^{(m)}}/q^{(m)}_{I_k^{(m)}+1}$. The proof is straightforward by using Lemma 4.

Lemma 5. *For any $\epsilon_q > 0$, there exists $n_2 > n_1$ such that, for all $n > n_2$*

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{1 \leq q^{(m)}_{I_k^{(m)}}/q^{(m)}_{I_k^{(m)}+1} < 1 + \epsilon_q\right\}\right] > 1 - \frac{2c_1}{n^2} \quad (24)$$

where the constant c_1 is the same as in Lemma 4.

Proof. Define an event $D_k \triangleq \left\{q^{(m)}_{I_k^{(m)}+1} < W_m q^\circ_{k+\lceil k^{1/2}\rceil}\right\}$. Noting that $\{d_i^{(m)}, i \geq I_k^{(m)} + 1\} \subseteq \{d_i^\circ,\, i \geq k+1\}$, we have $q^{(m)}_{I_k^{(m)}+1} < W_m q_k^\circ$, and hence the complement $D_k^c = \left\{W_m q^\circ_{k+\lceil k^{1/2}\rceil} < q^{(m)}_{I_k^{(m)}+1} < W_m q_k^\circ\right\}$. Using a similar approach to Lemma 4, we obtain $\mathbb{P}[D_k] = (1-\mu_m)^{\lceil k^{1/2}\rceil}$ and for all $n > n_1$

$$\mathbb{P}\left[\cap_{k\geq n}\mathcal{D}_k^c\right] = 1 - \mathbb{P}\left[\cup_{k\geq n}\mathcal{D}_k\right] \geq 1 - \sum_{k\geq n}\mathbb{P}\left[\mathcal{D}_k\right] \geq 1 - \frac{c_1}{n^2}.$$

which, togehter with (23), we obtain

$$\begin{aligned}\mathbb{P}\Bigg[\bigcap_{k\geq n}\Bigg\{&W_m q_k^\circ > q^{(m)}_{I_k^{(m)}+1} > W_m q^\circ_{k+\lceil k^{1/2}\rceil},\\ &W_m q^\circ_{k-\lceil k^{1/2}\rceil+1} > q^{(m)}_{I_k^{(m)}} > W_m q_k^\circ\Bigg\}\Bigg] > 1 - \frac{2c_1}{n^2},\end{aligned}$$

which implies

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{1 \leq \frac{q^{(m)}_{I^{(m)}_k}}{q^{(m)}_{I^{(m)}_k+1}} < \frac{q^\circ_{k-\lceil k^{1/2}\rceil+1}}{q^\circ_{k+\lceil k^{1/2}\rceil}}\right\}\right] > 1 - \frac{2c_1}{n^2}.$$

Using $q^\circ_{k-\lceil k^{1/2}\rceil+1}/q^\circ_{k+\lceil k^{1/2}\rceil} \to 1$, we finish the proof. □

Next, we establish the following lemma to estimate the functional relationship between $\left(\sum_{i=I^{(m)}_k+1}^\infty q^{(m)}_i\right)^{-1}$ and $\left(q^{(m)}_{I^{(m)}_k}\right)^{-1}$ based on Lemma 3 and Lemma 4.

Lemma 6. *For $\epsilon_p > 0$, there exists $n_3 > \max\{n_0, n_1\}$ and $c_2 > 0$ such that, for all $n > n_3$,*

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{(1-\epsilon_p)\left(\sum_{i=I^{(m)}_k+1}^{\infty} q^{(m)}_i\right)^{-1} < \tilde{\Phi}_m\left(\left(q^{(m)}_{I^{(m)}_k}\right)^{-1}\right) < (1+\epsilon_p)\left(\sum_{i=I^{(m)}_k+1}^{\infty} q^{(m)}_i\right)^{-1}\right\}\right] > 1 - \frac{c_2}{n^2}, \quad (25)$$

where

$$\tilde{\Phi}_m(x) = (W_m c_\circ)^{-1/\alpha_\circ}\mu_m^{-1}x^{1-1/\alpha_\circ}(\alpha_\circ - 1),\ \alpha_\circ > 1. \quad (26)$$

Proof. Since $q^\circ_k \sim c_\circ/k^{\alpha_\circ}$, there exist constants $\epsilon_k > 0, k \geq 1$ satisfying $\lim_{k\to\infty}\epsilon_k \to \infty$ such that

$$q^\circ_k > \frac{(1-\epsilon_k)c_\circ}{k^{\alpha_\circ}},\ q^\circ_{k-\lceil k^{1/2}\rceil+1} < \frac{(1+\epsilon_k)c_\circ}{(k-\lceil k^{1/2}\rceil+1)^{\alpha_\circ}}. \quad (27)$$

Combining (20) and (27), we obtain, for $n > n_1$

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{\frac{\left(k-\lceil k^{1/2}\rceil+1\right)^{\alpha_\circ}}{(1+\epsilon_k)W_m c_\circ} < \left(q^{(m)}_{I^{(m)}_k}\right)^{-1} < \frac{k^{\alpha_\circ}}{(1-\epsilon_k)W_m c_\circ}\right\}\right] \geq 1 - \frac{c_1}{n^2}. \quad (28)$$

Noting that $\{d^{(m)}_i, i \leq I^{(m)}_k\} \subseteq \{d^\circ_i,\, i \leq k\}$ and $\{d^{(m)}_i, i \geq I^{(m)}_k + 1\} \subseteq \{d^\circ_i,\, i \geq k+1\}$, we have

$$W_m Y^{(m)}_{n+1} = \sum_{i=n+1}^{\infty} W_m q^\circ_i X^{(m)}_i = \sum_{i=I^{(m)}_n+1}^{\infty} q^{(m)}_i. \quad (29)$$

Combining (29) and Lemma 3, we obtain, for $n > n_0$,

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{\left(W_m\left(\mathbb{E}\left[Y^{(m)}_{k+1}\right] + 1/k^{\alpha_\circ-\frac{2}{3}}\right)\right)^{-1} < \left(\sum_{i=I^{(m)}_k+1}^{\infty} q^{(m)}_i\right)^{-1} < \left(W_m\left(\mathbb{E}\left[Y^{(m)}_{k+1}\right] - 1/k^{\alpha_\circ-\frac{2}{3}}\right)\right)^{-1}\right\}\right] \geq 1 - \frac{c_0}{n^2}. \quad (30)$$

Since $\mathbb{E}\big[Y^{(m)}_{k+1}\big] = \mu_m\sum_{i=k+1}^\infty q^\circ_i \sim \mu_m c_\circ(\alpha_\circ-1)^{-1}(k+1)^{1-\alpha_\circ}$, there exist constants $\epsilon_{1,k} > 0$ with $\lim_{k\to\infty}\epsilon_{1,k} = 0$ such that

$$\frac{(1-\epsilon_{1,k})\mu_m c_\circ}{(\alpha_\circ-1)(k+1)^{\alpha_\circ-1}} < \mathbb{E}[Y^{(m)}_{k+1}] < \frac{(1+\epsilon_{1,k})\mu_m c_\circ}{(\alpha_\circ-1)(k+1)^{\alpha_\circ-1}}. \quad (31)$$

Combining (30) and (31), we have, there exist constants $\tau_{1,k}$, $\tau_{2,k}$ satisfying $\lim_{k\to\infty}\tau_{1,k}, \tau_{2,k} \to 1$ such that, for $n > n_0$,

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{\frac{(\alpha_\circ-1)k^{\alpha_\circ-1}}{\mu_m W_m c_\circ}\tau_{1,k} < \left(\sum_{i=I^{(m)}_k+1}^{\infty} q^{(m)}_i\right)^{-1} < \frac{(\alpha_\circ-1)k^{\alpha_\circ-1}}{\mu_m W_m c_\circ}\tau_{2,k}\right\}\right] > 1 - \frac{c_0}{n^2}, \quad (32)$$

Combining (26) and (28) implies, for $n > \max\{n_0, n_1\}$,

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{\frac{(\alpha_\circ-1)k^{\alpha_\circ-1}}{\mu_m W_m c_\circ}\delta_{1,k} < \tilde{\Phi}_m\left(\left(q^{(m)}_{I^{(m)}_k}\right)^{-1}\right) < \frac{(\alpha_\circ-1)k^{\alpha_\circ-1}}{\mu_m W_m c_\circ}\delta_{2,k}\right\}\right] > 1 - \frac{c_1}{n^2} \quad (33)$$

where constants $\delta_{1,k}$, $\delta_{2,k} \to 1$ as $k \to \infty$. Using a union bound to (32) and (33), we obtain, for $n > \max\{n_0, n_1\}$,

$$\mathbb{P}\left[\bigcap_{k\geq n}\left\{\frac{\delta_{1,k}}{\tau_{2,k}}\left(\sum_{i=I^{(m)}_k+1}^{\infty} q^{(m)}_i\right)^{-1} < \tilde{\Phi}_m\left(\left(q^{(m)}_{I^{(m)}_k}\right)^{-1}\right) < \frac{\delta_{2,k}}{\tau_{1,k}}\left(\sum_{i=I^{(m)}_k+1}^{\infty} q^{(m)}_i\right)^{-1}\right\}\right] > 1 - \frac{c_0+c_1}{n^2} \quad (34)$$

Since $\delta_{1,k}/\tau_{2,k}$, $\delta_{2,k}/\tau_{1,k} \to 1$ as $k \to \infty$, for any $\epsilon_p > 0$, there exists $n_3 > \max\{n_0, n_1\}$ such that, for any $n > n_3$, $1 - \epsilon_p < \delta_{1,k}/\tau_{2,k}, \delta_{2,k}/\tau_{1,k} < 1 + \epsilon_p$, which, together with (34), completes the proof. □

To use Lemma 1, we define $T_m(x) = \sum_{i=1}^\infty\big(1 - \big(1 - q^{(m)}_i\big)^x\big)$, which is equivalent to

$$T_m(x) = \sum_{i=1}^{\infty}\left(1 - (1 - W_m q^\circ_i)^x\right)X^{(m)}_i. \quad (35)$$

We now derive an approximation of $T_m(\cdot)$ using Lemma 7. On the proof, we first rewrite random variable $1/W_m = W_{m_1} + W_{m_2}$, where $W_{m_1} = \sum_{i=1}^n q^\circ_i X^{(m)}_i$ and $W_{m_2} = \sum_{i=n+1}^\infty q^\circ_i X^{(m)}_i$. For W_{m_1}, using Lemma 4, we show that for n large enough, $W_{m_2} \approx c(n)^{1-\alpha_\circ}$ with high probability. For W_{m_1}, note that, conditional on $\{H(d^\circ_i), 1 \leq i \leq n\}$, W_{m_1} is deterministic. Combining these two results, we have, for n large enough, conditional on $\{H(d^\circ_i), 1 \leq i \leq n\}$, W_m is deterministic. Based on this fact and conditional on $\{H(d^\circ_i), 1 \leq i \leq n\}$, we apply the Bernstein's inequality (12) for (35) and obtain the estimation (41). By noting that the bound in (41) is independent of the hashing function H, unconditional on $\{H(d^\circ_i), 1 \leq i \leq n\}$, we finish the proof.

Lemma 7. *For any $\epsilon_2 > 0$, there exists $x_0 > 0$, such that for all $x \geq x_0$,*

$$\mathbb{P}\left[\left\{(1-\epsilon_2)\mu_m\Gamma(1-1/\alpha_\circ)\left(c_\circ W_m x\right)^{\frac{1}{\alpha_\circ}} < T_m(x) < (1+\epsilon_2)\mu_m\Gamma(1-1/\alpha_\circ)\left(c_\circ W_m x\right)^{\frac{1}{\alpha_\circ}}\right\}\right] > 1 - \frac{c_3}{x^2}, \quad (36)$$

where c_3 is a positive constant.

Proof. For $n_x = \lfloor x^\sigma \rfloor$ with $\sigma < \alpha_0^{-1}$, we define $W_{m_1} = \sum_{i=1}^{n_x} q_i^\circ X_i^{(m)}$ and $W_{m_2} = \sum_{i=n_x+1}^{\infty} q_i^\circ X_i^{(m)}$. Recalling the definition of $Y_n^{(m)}$, we have $W_{m_2} = Y_{n_x+1}^{(m)}$, which, together with (18) and (31), we obtain, for $\epsilon > 0$, there exists a large $x_1 > 0$ such that for all $x > x_1$

$$\mathbb{P}\Big[(1-\epsilon)K(n_x) < W_{m_2} < (1+\epsilon)K(n_x)\Big] > 1 - 2\exp(-n_x^{1/3}v), \quad (37)$$

where $K(n_x) \triangleq \mu_m c_\circ(\alpha_\circ - 1)^{-1}(n_x+1)^{1-\alpha_\circ}$ and ν is the same constant as in (18). Noting $W_m = 1/(W_{m_1} + W_{m_2})$ and using (35) and (37), we have

$$\mathbb{P}\Big[I_1 + I_2 < T_m(x) < I_1 + I_3\Big] > 1 - 2\exp(-n_x^{\frac{1}{3}}v). \quad (38)$$

where $I_1 = \sum_{i=1}^{n_x} \left(1 - (1 - W_m q_i^\circ)^x\right) X_i^{(m)}$,

$$I_3 = \sum_{i=n_x+1}^{\infty} \left(1 - \left(1 - \frac{q_i^\circ}{W_{m_1} + (1-\epsilon)K(n_x)}\right)^x\right) X_i^{(m)}. \quad (39)$$

and I_2 is defined by replacing $(1-\epsilon)$ in I_3 with $(1+\epsilon)$.

Let $M_{n_x} \triangleq \{H(d_i^\circ) = S_{m_i}, 1 \le i \le n_x\}$ be an event that the first n_x data items are hashed to servers $S_{m_1}, \dots, S_{m_{n_x}}$. Let $p_i = q_i^\circ/(W_{m_1} + (1-\epsilon)K(n_x))$. From (35), we have $0 < W_m q_i^\circ < 1$ if $X_i^{(m)} = 1$ and $(1-(1-W_m q_i^\circ)^x)X_i^{(m)} = 0$ otherwise. Thus, without changing the expression of $T_m(x)$, we assume $0 < W_m q_i^\circ < 1$ for all $i \ge 1$. Note that $p_i \to W_m q_i^\circ$ as $n_x \to \infty$. Then, for n_x large enough and conditional on M_{n_x}, we have $p_i \sim c_1/i^{\alpha_\circ}$ and $0 < p_i < 1$, where the constant $c_1 = c_\circ/(W_{m_1} + (1-\epsilon)K(n_x))$. Similar to the derivation of Lemma 3, applying Lemma 2 to $I_3 x^{-1/\alpha_\circ} = \sum_{i=n_x+1}^{\infty} \left(1 - (1-p_i)^x\right) x^{-1/\alpha_\circ} X_i^{(m)}$ and using Lemma 1 in [9], we obtain, for $\forall\, \epsilon_1 > 0$, there exists a large x_2 such that for all $x > x_2$,

$$\mathbb{P}\Bigg[(1-\epsilon_1)\mu_m\Gamma(1-1/\alpha_\circ)\left(\frac{c_\circ x}{W_{1-\epsilon}}\right)^{\frac{1}{\alpha_\circ}} < I_3 < (1+\epsilon_1)\mu_m \Gamma(1-1/\alpha_\circ)\left(\frac{c_\circ x}{W_{1-\epsilon}}\right)^{\frac{1}{\alpha_\circ}} \Bigg| M_{n_x}\Bigg] > 1 - \frac{c_4}{x^2}, \quad (40)$$

where $W_{1-\epsilon} = W_{m_1} + (1-\epsilon)K(n_x)$ and c_4 is a positive constant. A similar result holds for I_2 by replacing $(1-\epsilon)$ and c_4 in (40) with $(1+\epsilon)$ and c_5, respectively. Recalling the definition (35) and conditional on the event M_{n_x}, we have $I_1 \le n_x$, which, in conjunction with $n_x < x^\sigma$ and $\sigma < 1/\alpha_\circ$, implies that $\lim_{x\to\infty} I_1/x^{\frac{1}{\alpha_\circ}} = 0$. Then, conditional on M_{n_x}, we have for $\forall\, \epsilon_2 > \epsilon_1$, there exists a sufficiently large x_3 such that for all $x > x_3$,

$$0 < I_1 < (\epsilon_2 - \epsilon_1)\mu_m\Gamma(1-1/\alpha_\circ)\left(c_\circ x/W_{1-\epsilon}\right)^{\frac{1}{\alpha_\circ}},$$

which, using (38) and (40) and a union bound, implies that for all $x > x_4 \triangleq \max\{x_1, x_2, x_3\}$,

$$\mathbb{P}\Bigg[(1-\epsilon_2)\mu_m\Gamma(1-1/\alpha_\circ)\left(\frac{c_\circ x}{W_{1+\epsilon}}\right)^{\frac{1}{\alpha_\circ}} < T_m(x) < (1+\epsilon_2) \mu_m\Gamma(1-1/\alpha_\circ)\left(\frac{c_\circ x}{W_{1-\epsilon}}\right)^{\frac{1}{\alpha_\circ}} \Bigg| M_{n_x}\Bigg] > 1 - \frac{c_3}{x^2}, \quad (41)$$

where $W_{1+\epsilon} = W_{m_1} + (1+\epsilon)K(n_x)$ and c_3 is a positive constant related to c_4, c_5 and ν. Note that $1 - c_3/n^2$ on the right side of (41) is independent of M_{n_x} and $W_{1+\epsilon}, W_{1-\epsilon} \to W_m$ as $x \to \infty$. Thus, for x large enough, unconditional on M_{n_x} and passing $\epsilon_1, \epsilon \to 0$, we obtain (36). □

Now, we use Lemmas 1, 5, 6 and 7 to prove Theorem 1 by applying the Borel-Cantelli lemma.

Proof of Theorem 1. For the two bounds proved in Lemma 5 and Lemma 6, it is easy to verify that $\sum_{n=1}^{\infty} (2c_1 + c_2)/n^2 < \infty$, which, by the Borel-Cantelli lemma, implies that, almost surely for each $H \in \mathcal{H}$, there exists a finite n_H, such that for all $n > n_H$, we have $1 \le q_{I_n^{(m)}}^{(m)}/q_{I_n^{(m)}+1}^{(m)} < 1 + \epsilon_q$ and $(1-\epsilon_p)\Big(\sum_{i=I_k^{(m)}+1}^{\infty} q_i^{(m)}\Big)^{-1} < \tilde{\Phi}_m\Big(\Big(q_{I_k^{(m)}}^{(m)}\Big)^{-1}\Big) < (1+\epsilon_p)\Big(\sum_{i=I_k^{(m)}+1}^{\infty} q_i^{(m)}\Big)^{-1}$, where $\tilde{\Phi}_m$ is defined in Lemma 6.

Now, note that x in Lemma 7 represents the cache size, which also takes integer values. For the bound in Lemma 7, we have $\sum_{x=1}^{\infty} c_3/x^2 < \infty$, which, by Borel-Cantelli lemma, implies that, almost surely for each $H \in \mathcal{H}$, there exists a finite x_H, such that for all $x > x_H$, we have $(1-\epsilon_2)\mu_m\Gamma(1-1/\alpha_\circ)\left(c_\circ W_m x\right)^{\frac{1}{\alpha_\circ}} < T_m(x) < (1+\epsilon_2)\mu_m\Gamma(1-1/\alpha_\circ)\left(c_\circ W_m x\right)^{\frac{1}{\alpha_\circ}}$.

These two facts, using Lemma 1 and passing $\epsilon_q, \epsilon_p, \epsilon_2 \to 0$, implies that, almost surely for all H,

$$\mathbb{P}\left[C_0 > x_m \big| J_0 = m, H\right] \sim \frac{(\mu_m\Gamma(1-1/\alpha_\circ))^{\alpha_\circ} c_\circ W_m}{\alpha_\circ x_m^{\alpha_\circ - 1}},$$

which finishes the proof of Theorem 1. □

B. Proof of Theorem 2

Proof. By Theorem 1, the miss ratio of cluster $\mathcal{C}$ satisfies

$$\mathbb{P}_{miss}^{\mathcal{C},H} = \sum_{m=1}^{N} \mathbb{P}\left[C_0 > x_m \big| J_0 = m, H\right]\mathbb{P}\left[J_0 = m \big| H\right] \sim \sum_{i=1}^{N} \frac{(\mu_m\Gamma(1-1/\alpha_\circ))^{\alpha_\circ} c_\circ}{\alpha_\circ b_m^{\alpha_\circ - 1} x^{\alpha_\circ - 1}},$$

which holds almost surely for all H. Noting that $q_i^\circ \sim c_\circ/i^{\alpha_\circ}$ and using Theorem 3 of [26], we can obtain $\mathbb{P}[C_0 > \bar{x}] \sim c_\circ(\Gamma(1-\alpha_\circ^{-1}))^{\alpha_\circ}/(\alpha_\circ \bar{x}^{\alpha_\circ - 1})$, which, in conjunction with (7), yields (6). □

C. Proof of Theorem 3

We first show that if $q_i^\circ \sim c_\circ/i^{\alpha_\circ}$, then, as $\bar{x} \to \infty$,

$$\mathbb{P}_{\text{CT}}\left[C_0 > \bar{x}\right] \sim \mathbb{P}\left[C_0 > \bar{x}\right]. \quad (42)$$

The proof is presented in the technical report [46]. Combining (42) and Theorem 2 yields Theorem 3.

REFERENCES

[1] D. Singh and C. K. Reddy, "A survey on platforms for big data analytics," *Journal of Big Data*, vol. 2, no. 1, p. 8, 2015.

[2] D. Karger, E. Lehman, T. Leighton, R. Panigrahy, M. Levine, and D. Lewin, "Consistent hashing and random trees: Distributed caching protocols for relieving hot spots on the world wide web," in *Proceedings of the Twenty-ninth Annual ACM Symposium on Theory of Computing*, ser. STOC '97, 1997, pp. 654–663.

[3] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels, "Dynamo: Amazon's highly available key-value store," in *Proceedings of Twenty-first ACM SIGOPS Symposium on Operating Systems Principles*, ser. SOSP '07. ACM, 2007, pp. 205–220.

[4] "Aerospike," http://www.aerospike.com/.

[5] "Memcached," http://memcached.org/.

[6] "Redis," http://redis.io/.

[7] A. S. Tanenbaum, *Modern Operating Systems*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2001.

[8] S. Vanichpun and A. M. Makowski, "The output of a cache under the independent reference model: where did the locality of reference go?" in *ACM SIGMETRICS Performance Evaluation Review*, vol. 32, no. 1. ACM, 2004, pp. 295–306.

[9] J. Tan, G. Quan, K. Ji, and N. Shroff, "On resource pooling and separation for LRU caching," in *Proceedings of the 2018 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science*. ACM, 2018.

[10] R. Nishtala, H. Fugal, S. Grimm, M. Kwiatkowski, H. Lee, H. C. Li, R. McElroy, M. Paleczny, D. Peek, P. Saab, D. Stafford, T. Tung, and V. Venkataramani, "Scaling Memcache at Facebook," in *Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13)*. Lombard, IL: USENIX, 2013, pp. 385–398.

[11] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson, *Introduction to Algorithms*, 2nd ed. McGraw-Hill Higher Education, 2001.

[12] D. Karger, A. Sherman, A. Berkheimer, B. Bogstad, R. Dhanidina, K. Iwamoto, B. Kim, L. Matkins, and Y. Yerushalmi, "Web caching with consistent hashing," in *Proceedings of the Eighth International Conference on World Wide Web*, ser. WWW '99, 1999, pp. 1203–1213.

[13] J. Wang, "A survey of web caching schemes for the Internet," *SIGCOMM Computer Communication Review*, vol. 29, no. 5, pp. 36–46, Oct. 1999.

[14] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan, "Chord: a scalable peer-to-peer lookup protocol for Internet applications," *IEEE/ACM Transactions on Networking (TON)*, vol. 11, no. 1, pp. 17–32, 2003.

[15] A. Lakshman and P. Malik, "Cassandra: a decentralized structured storage system," *ACM SIGOPS Operating Systems Review*, vol. 44, no. 2, pp. 35–40, 2010.

[16] T. Schickinger and A. Steger, "Simplified witness tree arguments," *Lecture notes in computer science*, pp. 71–87, 2000.

[17] S. Novakovic, P. Faraboschi, K. Keeton, R. Schreiber, E. Bugnion, and B. Falsafi, "Never mind networking: Using shared non-volatile memory in scale-out software," in *presentation, 2nd Int'l Workshop Rack-Scale Computing (WRSC 15)*, 2015.

[18] S. Edelkamp, "Planning with pattern databases," in *Sixth European Conference on Planning*, 2014.

[19] M. Dietzfelbinger and M. Rink, "Applications of a splitting trick," *Automata, Languages and Programming*, pp. 354–365, 2009.

[20] A. Pagh and R. Pagh, "Uniform hashing in constant time and optimal space," *SIAM Journal on Computing*, vol. 38, no. 1, pp. 85–96, 2008.

[21] Z. Drudi, N. J. Harvey, S. Ingram, A. Warfield, and J. Wires, "Approximating hit rate curves using streaming algorithms," in *LIPIcs-Leibniz International Proceedings in Informatics*, vol. 40. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2015.

[22] N. Gast and B. Van Houdt, "Transient and steady-state regime of a family of list-based cache replacement algorithms," *ACM SIGMETRICS Performance Evaluation Review*, vol. 43, no. 1, pp. 123–136, 2015.

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

[24] 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.

[25] P. R. Jelenković and A. Radovanović, "Least-recently-used caching with dependent requests," *Theoretical Computer Science*, vol. 326, no. 1-3, pp. 293–327, Oct. 2004.

[26] P. R. Jelenković, "Asymptotic approximation of the move-to-front search cost distribution and least-recently-used caching fault probabilities," *The Annals of Applied Probability*, no. 2, pp. 430–464, 1999.

[27] T. Osogami, "A fluid limit for a cache algorithm with general request processes," *Advances in Applied Probability*, vol. 42, no. 3, pp. 816–833, 2010.

[28] D. S. Berger, P. Gland, S. Singla, and F. Ciucu, "Exact analysis of TTL cache networks," in *32nd International symposium on Computer Performance, modeling, measurements, and evaluation (IFIP Performance'14)*, Turin, Italy, October 2014, pp. 2–23.

[29] M. Garetto, E. Leonardi, and V. Martina, "A unified approach to the performance analysis of caching systems," *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, vol. 1, no. 3, p. 12, 2016.

[30] C. Fricker, P. Robert, and J. Roberts, "A versatile and accurate approximation for LRU cache performance," in *Proceedings of the 24th International Teletraffic Congress*, ser. ITC '12, 2012, pp. 8:1–8:8.

[31] J. Roberts and N. Sbihi, "Exploring the memory-bandwidth tradeoff in an information-centric network," in *Teletraffic Congress (ITC), 2013 25th International*. IEEE, 2013, pp. 1–9.

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

[33] S. Ioannidis and E. Yeh, "Adaptive caching networks with optimality guarantees," in *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science*. ACM, 2016, pp. 113–124.

[34] D. S. Berger, P. Gland, S. Singla, and F. Ciucu, "Exact analysis of TTL cache networks," *Performance Evaluation*, vol. 79, pp. 2–23, 2014.

[35] N. C. Fofack, P. Nain, G. Neglia, and D. Towsley, "Performance evaluation of hierarchical TTL-based cache networks," *Computer Networks*, vol. 65, pp. 212 – 231, 2014.

[36] J. L. Carter and M. N. Wegman, "Universal classes of hash functions," in *Proceedings of the Ninth Annual ACM Symposium on Theory of Computing*, ser. STOC '77. New York, NY, USA: ACM, 1977, pp. 106–112.

[37] M. N. Wegman and J. L. Carter, "New classes and applications of hash functions," in *20th Annual Symposium on Foundations of Computer Science (sfcs 1979)*, Oct 1979, pp. 175–182.

[38] M. Mitzenmacher and S. Vadhan, "Why simple hash functions work: Exploiting the entropy in a data stream," in *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '08. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2008, pp. 746–755.

[39] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: evidence and implications," in *Proceedings of the 18th Conference on Information Communications*, 1999.

[40] C. Fricker, P. Robert, J. Roberts, and N. Sbihi, "Impact of traffic mix on caching performance in a content-centric network," in *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*. IEEE, 2012, pp. 310–315.

[41] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM, 2007, pp. 1–14.

[42] P. R. Jelenković and X. Kang, "LRU caching with moderately heavy request distributions," in *2007 Proceedings of the Fourth Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*. SIAM, 2007, pp. 212–222.

[43] G. Quan, K. Ji, and J. Tan, "LRU caching with dependent competing requests," in *2018 IEEE Conference on Computer Communications (INFOCOM)*, Honolulu, HI, April 2018.

[44] C. Berthet, "Approximation of LRU caches miss rate: Application to power-law popularities," *arXiv preprint arXiv:1705.10738*, 2017.

[45] J. Fill, "An exact formula for the move-to-front rule for self-organizing lists," *Journal of Theoretical Probability*, vol. 9, no. 1, pp. 113–160, 1996.

[46] K. Ji, G. Quan, and J. Tan, "Asymptotic miss ratio of LRU caching with consistent hashing," *arXiv preprint arXiv:1801.02436*, 2018.

[47] F. R. Chung and L. Lu, *Complex graphs and networks*. American mathematical society Providence, 2006, vol. 107.