TECHNICAL RESEARCH REPORT

The output of a cache under the independent reference model - Where did the locality of reference go?

by Sarut Vanichpun, Armand M. Makowski

CSHCN TR 2004-7
(ISR TR 2004-11)

The output of a cache under the independent reference model – Where did the locality of reference go?

ABSTRACT

We consider a cache operating under a demand-driven replacement policy when document requests are modeled according to the Independent Reference Model (IRM). We characterize the popularity pmf of the stream of misses from the cache, the so-called output of the cache, for a large class of cache replacement policies, including standard on-demand replacement algorithms such as the policy A_0 and the random policy, as well as the LRU and CLIMB policies. We measure strength of locality of reference in a stream of requests through the skewness of its popularity distribution. Using the notion of majorization to capture the degree of skewness, we show that for the policy A_0 and the random policy, the output always has less locality of reference than the input. However, we show by counterexamples that this is not always the case under the LRU and CLIMB policies when the input is selected according to a Zipf-like pmf. In that case, conjectures are offered (and supported by simulations) as to when LRU or CLIMB caching indeed reduces locality of reference.

1. INTRODUCTION

1.1 Cache hierarchies and transformations

Since its inception, the World Wide Web has seen an exponential increase in the number of its users and in the volume of objects to be accessed. This trend, which is not likely to abate anytime soon, is challenging cache architectures to meet the complementary mandates of *speed*, *scalability* and *reliability* which are central to delivering a satisfactory user experience.

Generally speaking, scalability requires some form of *hierarchical* organization. In the context of Web caching, this notion has led naturally to cache sharing through the deployment of *multi-layered* systems of *interconnected* caches which may be organized in a tree-like hierarchy or in more complicated meshes [6, 13] (and references therein).

Even a cursory review of the literature [3, 22] (and references therein) already reveals the large number of difficult and challenging issues that need to be addressed in order to ensure proper operations of these distributed multi-level caching systems [3]. However, lacking generally in most of the work done thus far, is a clear recognition of the system-wide nature of Web caching, whereby local *transformative* actions shape the streams of requests as they pass through successive caches. Recent exceptions can be found in the works [7, 12, 14] for cache management and in the references [18, 23, 24] for Web traffic analysis.

A framework has been introduced recently in [15] to describe the operation of Web caches in terms of transformations on streams of requests: The transformations of interest are decomposed into basic building blocks. Three *primitive* operations on streams of requests are identified, namely filtering, aggregation and disaggregation. From the point of view of caching, the Web can now be represented as an interconnected collection of nodes, each such node being a site where exactly one of the three primitive operations is executed. This allows for a structured representation of users, intermediaries (e.g., institutional and regional caches) and servers by appropriately concatenating these primitives.

In this context, *filtering* is identified as the transformation which produces the streams of misses at the output of a cache. Misses produced at a cache at some level in the hierarchy contribute to the stream of requests that gets directed to a cache at the next level or to a server. Given its ubiquity, it is natural to wonder as to the impact of this filtering operation, i.e., how are the statistics of the filtered stream related to those of the input stream? This is the issue we start addressing in this paper.

1.2 Locality of reference

The performance of any form of caching is determined by a number of factors, chief amongst them are the statistical properties of the streams of requests made to the cache. One important such property is the *locality of reference* present in a request stream whereby bursts of references are made in the near future to objects referenced in the recent past. The implications for cache management should be clear – Increased locality of reference should yield performance improvements for demand-driven caching that exploits recency of reference. Moreover, it is widely believed that good cache replacement strategies produce an output stream of requests exhibiting *less* locality of reference than the input stream of requests. In the context of multi-level caching, this reduction property is often perceived as one of the main reasons for why caching looses its effectiveness after some point in the hierarchy of caches. The main objective of this paper is to start a formal investigation of this reduction property in a simple framework.

The notion of locality and its importance for caching were first recognized by Belady [4] in the context of computer memory. Subsequently, a number of studies have shown that request streams for Web objects exhibit strong locality of reference[1] [17, 19]. Attempts at characterization were made early on by Denning through the working set model [11]. Yet, like the notion of burstiness used in traffic modeling, locality of reference, while endowed with a clear intuitive content, admits no simple definition. Not surprisingly, in spite of numerous efforts, no consensus has been reached on how to formalize the notion, let alone *compare* streams of requests on the basis of their locality of reference.[2]

It is by now widely accepted that the two main contributors to locality of reference are *temporal correlations* in the streams of

[1] At least in the short timescales

[2] An exception can be found in a recent paper by Fonseca et al [15].

requests and the *popularity distribution* of requested objects. To describe these two sources of locality, we assume the following generic setup which is used throughout the paper: We consider a universe of N cacheable items or documents, labeled $i = 1, \ldots, N$, and we write $\mathcal{N} = \{1, \ldots, N\}$. The successive requests arriving at the cache are modeled by a sequence $\{R_t,\ t = 0, 1, \ldots\}$ of $\mathcal{N}$-valued rvs.

The popularity of the sequence of requests $\{R_t,\ t = 0, 1, \ldots\}$ is defined as the pmf $\boldsymbol{p} = (p(i), \ldots, p(N))$ on $\mathcal{N}$ given by

$$p(i) := \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbf{1}\left[R_\tau = i\right] \quad a.s., \quad i = 1, \ldots, N, \qquad (1)$$

whenever these limits exist (and they do in most models treated in the literature). Temporal correlations are more delicate to define due to the "categorical" nature of the requests $\{R_t,\ t = 0, 1, \ldots\}$. The literature contains several metrics to do this, e.g., the inter-reference time [15, 17], the working set size [11] and the stack distance [1].

In this paper we focus exclusively on popularity. To isolate its contributions, we consider the situation where there is *no* temporal correlations in the stream of requests as would be the case under the *Independence Reference Model* (irm). More precisely, under the IRM with popularity pmf $\boldsymbol{p} = (p(1), \ldots, p(N))$, the successive requests $\{R_t,\ t = 0, 1, \ldots\}$ form a sequence of i.i.d. $\mathcal{N}$-valued rvs, each distributed according to the pmf $\boldsymbol{p}$.[3] Here, even in the absence of temporal correlations, locality of reference is present, in that the *skewness* of $\boldsymbol{p}$ acts as an indicator of the strength of locality of reference present in the stream, under the intuition that the more "balanced" the pmf $\boldsymbol{p}$, the weaker the locality of reference.[4]

1.3 Contributions

We consider a model of a cache operating under a demand-driven replacement policy when document requests are modeled according to the Independent Reference Model (IRM) [Section 2]. For this model we characterize the popularity pmf of the stream of misses from the cache in terms of the popularity pmf of the input [Section 3]. This characterization holds for a large class of cache replacement policies, including the optimal policy A_0, the random policy, the LRU policy and the CLIMB policy.[5]

We measure strength of locality of reference in a stream of requests by the skewness of its popularity distribution. As was done in [21], the degree of skewness in the popularity pmf is captured formally through the notion of *majorization (ordering)*. Basic facts concerning majorization [Section 4] enable us to develop generic comparison results between popularity pmfs of the input and output [Section 5].

We apply these results to the random policy [Section 6] and to the policy A_σ [Section 7]. In both instances, we conclude that the output pmf $\boldsymbol{p}^\star$ always has less locality of reference than the input. However, an arbitrary policy does not necessarily exhibit this reduction property in *all* circumstances, i.e., for an *arbitrary* pmf profile for the input and for *arbitrary* cache sizes. Surprisingly enough, even the popular LRU policy [Section 9] and its close relative, the CLIMB policy [Section 10], can fail to reduce locality of reference. For each of these policies, we explore the matter through counterexamples developed when the popularity pmf of the input is a Zipf-like pmf [Section 8]. For this class of input pmfs, we identify a condition involving the cache size and the number of cacheable documents under which reduction fails to occur at large enough values of the skewness parameter of the Zipf-like pmf. Under this condition, which we expect to be satisfied in practice, we show in effect that the output pmf $\boldsymbol{p}^\star$ may not exhibit less locality of reference than the input pmf $\boldsymbol{p}$ when the latter has too much of it to begin with. Various proofs are given in Appendix.

Simulations were carried out with Zipf-like pmfs as the input, and conjectures are offered as to when LRU or CLIMB caching indeed reduces locality of reference. All indications point to the possibility that for small enough cache sizes, the desired comparison of $\boldsymbol{p}$ and $\boldsymbol{p}^\star$ will hold; this will be the subject of future investigation. Finally, we believe that the framework developed can be applied not only to an IRM but also a general input model with correlations.

[3] Thus, $\mathbf{P}\left[R_t = i\right] = p(i)$ $(i = 1, \ldots, N)$ for all $t = 0, 1, \ldots$ and (1) holds with the given pmf $\boldsymbol{p}$ as seen in the next section.

[4] This is best appreciated by comparing the limiting cases $\boldsymbol{p} = (1-\delta, \varepsilon, \ldots, \varepsilon)$ (with $\delta = (N-1)\varepsilon \ll 1$) and the uniform popularity pmf $\boldsymbol{p}$ (with $p(1) = \ldots = p(N) = \frac{1}{N}$).

[5] The results are valid for a much larger class of policies, including random on-demand replacement algorithms [21] which generalize the policy A_0 and the random policy.

2. DEMAND-DRIVEN CACHING

Consider a universe $\mathcal{N}$ of N cacheable documents, say $\mathcal{N} := \{1, \ldots, N\}$. The system is composed of a server where a copy of each of these N documents is available, and of a cache of size M $(1 \leq M < N)$. Documents are first requested at the cache: If the requested document has a copy already in cache (i.e., a hit), this copy is downloaded from the cache by the user. If the requested document is not in cache (i.e., a miss), a copy is requested instead from the server to be put in the cache. If the cache is already full, then a document already in cache is evicted to make place for the copy of the document just requested. A *demand-driven* cache replacement policy (to be specified shortly) is assumed to be in use.

Consecutive user requests are modeled by a sequence of $\mathcal{N}$-valued rvs $\{R_t,\ t = 0, 1, \ldots\}$. For simplicity we say that request R_t occurs at time t $(t = 0, 1, \ldots)$. Let S_t denote the cache just before time t so that S_t is a subset of $\mathcal{N}$ with at most M elements, and let U_t denote the decision to be performed according to the cache replacement policy in force. Under a demand-driven cache replacement policy, the cache evolves according to [6]

$$S_{t+1} = \begin{cases} S_t & \text{if } R_t \in S_t \\ S_t + R_t & \text{if } R_t \notin S_t, |S_t| < M \\ S_t - U_t + R_t & \text{if } R_t \notin S_t, |S_t| = M \end{cases} \qquad (2)$$

for all $t = 0, 1, \ldots$, where $|S_t|$ denotes the cardinality of the set S_t. These dynamics reflect the following operational assumptions: (i) a requested document not in cache is *always* added to the cache if the cache is not full; and (ii) eviction is *mandatory* if the request R_t is not in cache S_t and the cache S_t is full, i.e., $|S_t| = M$.

The sequence of successive decisions $\{U_t,\ t = 0, 1, \ldots\}$ are determined through an eviction policy. Examples will be discussed later in the paper.

Throughout, we assume that the stream of requests $\{R_t,\ t = 0, 1, \ldots\}$ is modeled according to the standard *Independence Reference Model* (IRM) with popularity vector $\boldsymbol{p} = (p(1), \ldots, p(N))$. To avoid uninteresting situations, it is *always* the case that

$$p(i) > 0, \quad i = 1, \ldots, N. \qquad (3)$$

A pmf $\boldsymbol{p}$ on $\{1, \ldots, N\}$ satisfying (3) is said to be *admissible*.

[6] Throughout, for any subset S of $\{1, \ldots, N\}$, any element u in S and any element x in $\{1, \ldots, N\}$, we write $S - u + x$ to denote the subset of $\{1, \ldots, N\}$ obtained from S by removing u and then adding x to it, *in that order*.

Under this non-triviality condition (3), every document will eventually be requested as we note

$$\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[R_\tau = i\right] = p(i) > 0 \quad a.s.$$

by the Strong Law of Large Numbers. Thus, as we have in mind to study long term characteristics under demand-driven replacement policies, there is no loss of generality in assuming (as we do from now on) that the cache is full in that $|S_t| = M$ for all $t = 0, 1, \ldots$, and (2) simplifies to

$$S_{t+1} = \begin{cases} S_t & \text{if } R_t \in S_t \\ S_t - U_t + R_t & \text{if } R_t \notin S_t \end{cases} \qquad t = 0, 1, \ldots. \quad (4)$$

We close this section with some notation that will be used repeatedly: Let $\Lambda(M;\mathcal{N})$ be the collection of all *unordered* subsets of size M of $\mathcal{N} = \{1,\ldots,N\}$, and let $\Lambda^\star(M;\mathcal{N})$ be the collection of all *ordered* sequences of M *distinct* elements from $\mathcal{N}$. We write $\{i_1,\ldots,i_M\}$ (resp. $(i_1,\ldots,i_M)$) to denote an element in $\Lambda(M;\mathcal{N})$ (resp. $\Lambda^\star(M;\mathcal{N})$). For each $i = 1,\ldots,N$, let $\Lambda_i(M;\mathcal{N})$ (resp. $\Lambda_i^\star(M;\mathcal{N})$) denote the set of elements in $\Lambda(M;\mathcal{N})$ (resp. $\Lambda^\star(M;\mathcal{N})$) which do *not* contain i, i.e.,

$$\Lambda_i(M;\mathcal{N}) := \{s = \{i_1,\ldots i_M\} \in \Lambda(M;\mathcal{N}) : \ i \notin s\}$$

and

$$\Lambda_i^\star(M;\mathcal{N}) := \{s = (i_1,\ldots i_M) \in \Lambda^\star(M;\mathcal{N}) : \ i \notin s\}.$$

Lastly, for each $M = 1,\ldots,N$, the *elementary symmetric* function $E_{M,N} : \mathbf{R}^N \to \mathbf{R}$ is defined [20, p. 78] by

$$E_{M,N}(\boldsymbol{x}) := \sum_{\{i_1,\ldots,i_M\}\in\Lambda(M;\mathcal{N})} x_{i_1}\cdots x_{i_M}, \quad \boldsymbol{x} \in \mathbf{R}^N. \quad (5)$$

By convention we write $E_{0,N}(\boldsymbol{x}) = 1$ for all $\boldsymbol{x}$ in $\mathbf{R}^N$.

3. THE OUTPUT OF A CACHE

3.1 Definitions

Under the demand-driven caching operation (4), the output of the cache is the sequence of requests that incur a miss, i.e., when the incoming request cannot find the desired document in the cache. More precisely, a miss occurs at time t if R_t is *not* in S_t. Thus, we define recursively the time indices $\{\nu_k,\ k = 0, 1, \ldots\}$ by

$$\nu_0 = 0; \quad \nu_{k+1} := \nu_k + \mu_{k+1}, \quad k = 0, 1, \ldots$$

and

$$\mu_{k+1} := \inf\{\ell = 1, 2, \ldots : \ R_{\nu_k+\ell} \notin S_{\nu_k+\ell}\}$$

with the convention $\mu_{k+1} = \infty$ if either $\nu_k = \infty$ or if ν_k is finite but the set of indices entering the definition of μ_{k+1} is empty. With Δ denoting an element *not* in $\mathcal{N}$, we define the output process $\{R_k^\star,\ k = 0, 1, \ldots\}$ simply as

$$R_k^\star := \begin{cases} R_{\nu_k} & \text{if } \nu_k < \infty \\ \Delta & \text{if } \nu_k = \infty \end{cases}$$

for each $k = 1, 2, \ldots$. The requests $\{R_k^\star, k = 1, 2, \ldots\}$ are those requests among $\{R_t, t = 0, 1, \ldots\}$ which incur a miss and which get forwarded to the server (or to the higher level cache in a hierarchical Web caching system).

The statistics of the output stream $\{R_k^\star,\ k = 0, 1, \ldots\}$ are determined by the statistics of the input stream $\{R_t, t = 0, 1, \ldots\}$ and of the cache replacement policy in use. We are interested in evaluating the popularity vector $\boldsymbol{p}^\star = (p^\star(1),\ldots,p^\star(N))$ defined by

$$p^\star(i) := \lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K}\mathbf{1}\left[R_k^\star = i\right] \quad a.s. \quad (6)$$

for each $i = 1, 2, \ldots, N$, whenever these limits exist.

3.2 Finding $\boldsymbol{p}^\star$

The remainder of this section is devoted to establishing the existence and form of these limits. We do so under the assumption that the a.s. limit

$$\pi(s;\boldsymbol{p}) = \lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[S_\tau = s\right] \quad a.s. \quad (7)$$

exists for each s in $\Lambda(M;\mathcal{N})$, and is independent of the initial condition S_0. As we shall see shortly, this is a mild assumption which is satisfied under all eviction policies of interest considered in the literature. The basic result is contained in

THEOREM 1. *Assume the existence of the limits (7). For each $i = 1,\ldots,N$, the limit (6) exists and is given by*

$$\begin{aligned} p^\star(i) &= \lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K}\mathbf{1}\left[R_k^\star = i\right] \\ &= \frac{p(i)m(i;\boldsymbol{p})}{\sum_{j=1}^{N} p(j)m(j;\boldsymbol{p})} \quad a.s. \end{aligned} \quad (8)$$

where we have set

$$m(i;\boldsymbol{p}) = \sum_{s\in\Lambda_i(M;\mathcal{N})}\pi(s;\boldsymbol{p}). \quad (9)$$

Before giving a proof of Theorem 1, we note that the existence of the limits (7) implies

$$\begin{aligned} m(i;\boldsymbol{p}) &= \sum_{s\in\Lambda_i(M;\mathcal{N})}\left(\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[S_\tau = s\right]\right) \\ &= \lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\sum_{s\in\Lambda_i(M;\mathcal{N})}\mathbf{1}\left[S_\tau = s\right] \\ &= \lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[i \notin S_\tau\right] \quad a.s. \end{aligned} \quad (10)$$

for each $i = 1,\ldots,N$, and $m(i;\boldsymbol{p})$ thus represents the fraction of times that document i will not be in the cache. This quantity is determined by the popularity vector $\boldsymbol{p}$ of the input to the cache and by the eviction policy in use.

Proof. For each $t = 1, 2, \ldots$, let $K(t)$ denote the total number of misses up to time t at times $\tau = 1,\ldots,t$. Obviously, we have

$$K(t) := \sum_{\tau=1}^{t}\mathbf{1}\left[R_\tau \notin S_\tau\right]. \quad (11)$$

Fix $i = 1,\ldots,N$. We note that

$$\sum_{k=1}^{K(t)}\mathbf{1}\left[R_k^\star = i\right] = \sum_{\tau=1}^{t}\mathbf{1}\left[i \notin S_\tau\right]\mathbf{1}\left[R_\tau = i\right]$$

$$= \quad p(i)\sum_{\tau=1}^{t}\mathbf{1}\left[i\notin S_\tau\right] \qquad (12)$$

$$+\sum_{\tau=1}^{t}\mathbf{1}\left[i\notin S_\tau\right]\left(\mathbf{1}\left[R_\tau=i\right]-p(i)\right).$$

Invoking Rajchman's version of the Strong Law of Large Numbers [8, Thm. 5.1.2., p. 103], we find

$$\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[i\notin S_\tau\right]\left(\mathbf{1}\left[R_\tau=i\right]-p(i)\right)=0 \quad a.s. \qquad (13)$$

Next, combining (10) and (13), we get with the help of (12) that

$$\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[i\notin S_\tau\right]\mathbf{1}\left[R_\tau=i\right]$$
$$= \quad p(i)\sum_{s\in\Lambda_i(M;\mathcal{N})}\pi(s;\boldsymbol{p}) \quad a.s. \qquad (14)$$

Using the basic identity

$$K(t)=\sum_{\tau=1}^{t}\mathbf{1}\left[R_\tau\notin S_\tau\right]=\sum_{i=1}^{N}\sum_{\tau=1}^{t}\mathbf{1}\left[i\notin S_\tau\right]\mathbf{1}\left[R_\tau=i\right]$$

for each $t=1,2,\ldots$, we conclude from (14) that

$$\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[R_\tau\notin S_\tau\right]$$
$$= \quad \sum_{i=1}^{N}\left(\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[i\notin S_\tau\right]\mathbf{1}\left[R_\tau=i\right]\right)$$
$$= \quad \sum_{i=1}^{N}p(i)\left(\sum_{s\in\Lambda_i(M;\mathcal{N})}\pi(s;\boldsymbol{p})\right) \quad a.s. \qquad (15)$$

It is now immediate that the following limit exists a.s. independently of the initial conditions, and is given by

$$\lim_{t\to\infty}\frac{1}{K(t)}\sum_{k=1}^{K(t)}\mathbf{1}\left[R_k^\star=i\right]$$
$$= \quad \frac{\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[R_\tau\notin S_\tau\right]\mathbf{1}\left[R_\tau=i\right]}{\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=1}^{t}\mathbf{1}\left[R_\tau\notin S_\tau\right]}$$
$$= \quad \frac{p(i)m(i;\boldsymbol{p})}{\sum_{j=1}^{N}p(j)m(j;\boldsymbol{p})} \quad a.s. \qquad (16)$$

The desired conclusion is readily obtained from (16) once we observe the convergence $\lim_{t\to\infty}K(t)=\infty$ a.s. monotonically so that the sequence $\{K(t),\ t=1,2,\ldots\}$ a.s. exhausts $\mathbb{N}$, and the a.s. existence of the limit in (16) implies the a.s. existence of the limit (6) with limiting value (8)-(9). ■

3.3 Remarks

The existence of the limits (7) is often validated as follows: For most eviction policies considered in the literature, the dynamics of the cache can be characterized through the evolution of suitably defined variables $\{\Omega_t,\ t=0,1,\ldots\}$ where Ω_t is known as the *state of the cache* at time t. The cache state is selected such that (i) the eviction decision U_t at time t can be expressed as a function of the past $(\Omega_0,R_0,U_0,\ldots,\Omega_{t-1},R_{t-1},U_t,\Omega_t,R_t)$; and (ii) the set S_t of documents in the cache at time t can be recovered from Ω_t.

For instance, under the random policy [Section 6] and the policies A_σ and A_0 [Section 7], we can take the cache state to be the (unordered) set of documents in the cache, hence the cache state is an element of $\Lambda(M;\mathcal{N})$ and $\Omega_t=S_t$ for all $t=0,1,\ldots$.

For the policies LRU and CLIMB [Sections 9 and 10], the cache state is an element of $\Lambda^\star(M;\mathcal{N})$ and Ω_t is a permutation of the elements in S_t for all $t=0,1,\ldots$. For these policies, under the IRM, the sequence of rvs $\{(\Omega_t,R_t),\ t=0,1,\ldots\}$ form a time-homogeneous Markov chain over a finite state space, and standard ergodic results for finite state Markov chains readily yield the existence of the limits (7).

4. MAJORIZATION – A PRIMER

The concept of *majorization* [20] provides a powerful tool to formalize statements concerning the relative skewness in the components of two vectors, viz., the components $(x_1,\ldots,x_N)$ of the vector $\boldsymbol{x}$ are "more spread out" or "more balanced" than the components $(y_1,\ldots,y_N)$ of the vector $\boldsymbol{y}$: For vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathbb{R}^N$, we say that $\boldsymbol{x}$ is *majorized* by $\boldsymbol{y}$, and write $\boldsymbol{x}\prec\boldsymbol{y}$, whenever the conditions

$$\sum_{i=1}^{n}x_{[i]}\le\sum_{i=1}^{n}y_{[i]}, \quad n=1,2,\ldots,N-1 \qquad (17)$$

and

$$\sum_{i=1}^{N}x_i=\sum_{i=1}^{N}y_i \qquad (18)$$

hold with $x_{[1]}\ge x_{[2]}\ge\ldots\ge x_{[N]}$ and $y_{[1]}\ge y_{[2]}\ge\ldots\ge y_{[N]}$ denoting the components of $\boldsymbol{x}$ and $\boldsymbol{y}$ arranged in decreasing order, respectively.

As elegantly demonstrated in the monograph of Marshall and Olkin [20], this notion has found widespread use in many diverse branches of mathematics and their applications, viz. in computer databases [10] and storage [25].

We begin with a sufficient condition for majorization which is extracted from the discussion in [20, B.1, p. 129].

PROPOSITION 1. *Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be distinct elements of $\mathbb{R}^N$ such that*

$$\sum_{i=1}^{N}x_i=\sum_{i=1}^{N}y_i. \qquad (19)$$

Whenever, $x_1\ge x_2\ge\ldots\ge x_N$, if there exists some $k=1,\ldots,N-1$ such that $x_i\le y_i$, $i=1,\ldots,k$, and $x_i\ge y_i$, $i=k+1,\ldots,N$, then the comparison $\boldsymbol{x}\prec\boldsymbol{y}$ holds.

The following sufficient condition for majorization will be useful in the sequel. It was already announced in [20, B.1.b, p. 129] without proof.

THEOREM 2. *Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be distinct elements of $\mathbb{R}^N$ such that (19) holds. Whenever $x_1\ge x_2\ge\ldots\ge x_N>0$, and the ratios $\frac{y_i}{x_i}$, $i=1,\ldots,N$, are decreasing in i, we have the comparison $\boldsymbol{x}\prec\boldsymbol{y}$.*

Proof. Under the condition $x_i>0$, $i=1,\ldots,N$, we find that (19) can be rewritten as

$$\sum_{i=1}^{N}x_i\left(\frac{y_i}{x_i}-1\right)=0. \qquad (20)$$

If the ratios $\frac{y_i}{x_i}$, $i = 1, \ldots, N$, are decreasing in i, then by virtue of (20) there must exist some k with $1 \leq k < N$ such that

$$\frac{y_i}{x_i} - 1 \geq 0, \quad i = 1, \ldots, k$$

and

$$\frac{y_i}{x_i} - 1 \leq 0, \quad i = k+1, \ldots, N.$$

In other words, $x_i \leq y_i$ for $i = 1, \ldots, k$ and $y_i \leq x_i$ for $i = k+1, \ldots, N$. We now readily obtain the comparison $\boldsymbol{x} \prec \boldsymbol{y}$ by applying Proposition 1. ■

With any element of $\mathbb{R}^N$ such that $\sum_{i=1}^N x_i \neq 0$, we associate the *normalized* vector $\bar{\boldsymbol{x}}$ as the element of $\mathbb{R}^N$ defined by

$$\bar{\boldsymbol{x}} := (\sum_{i=1}^N x_i)^{-1}(x_1, \ldots, x_N).$$

With this notation we can now present a useful corollary to Theorem 2.

COROLLARY 1. *Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be distinct elements of $\mathbb{R}^N$ such that $\sum_{i=1}^N y_i > 0$. Whenever $x_1 \geq x_2 \geq \ldots \geq x_N > 0$, and the ratios $\frac{y_i}{x_i}$, $i = 1, \ldots, N$, are decreasing in i, we have the comparison $\bar{\boldsymbol{x}} \prec \bar{\boldsymbol{y}}$.*

Proof. Under the enforced assumptions, we note the inequalities $\sum_{i=1}^N x_i > 0$ and $\bar{x}_1 \geq \bar{x}_2 \geq \ldots \geq \bar{x}_N > 0$ with the ratios $\frac{\bar{y}_i}{\bar{x}_i}$, $i = 1, \ldots, N$, decreasing in i. Obviously, $\sum_{i=1}^N \bar{x}_i = \sum_{i=1}^N \bar{y}_i = 1$ and we get the desired result when applying Theorem 2 to $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$. ■

The following reformulation of Corollary 1 is used in the sequel.

LEMMA 1. *Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be distinct elements of $\mathbb{R}^N$ such that $x_i > 0$, $i = 1, \ldots, N$ and $\sum_{i=1}^N y_i > 0$. If*

$$\frac{y_i}{x_i} \geq \frac{y_j}{x_j} \tag{21}$$

whenever $x_i \geq x_j$ for distinct $i, j = 1, \ldots, N$, then the comparison $\bar{\boldsymbol{x}} \prec \bar{\boldsymbol{y}}$ holds.

Before giving a proof, we introduce the following notation: Let σ denote a permutation of $\{1, \ldots, N\}$. With any element $\boldsymbol{x}$ in $\mathbb{R}^N$, we associate the *permuted* vector $\sigma(\boldsymbol{x})$ in $\mathbb{R}^N$ defined through

$$\sigma(\boldsymbol{x})_i = x_{\sigma(i)}, \quad i = 1, \ldots, N.$$

It is plain from the definition of majorization that for vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathbb{R}^N$, we have $\boldsymbol{x} \prec \boldsymbol{y}$ if and only if $\sigma(\boldsymbol{x}) \prec \boldsymbol{y}$ for any permutation σ of $\{1, \ldots, N\}$.

Proof. Let σ denote a permutation of $\{1, \ldots, N\}$ such that $x_{\sigma(1)} \geq x_{\sigma(2)} \geq \ldots \geq x_{\sigma(N)}$. The enforced monotonicity assumptions can be restated as

$$\frac{y_{\sigma(1)}}{x_{\sigma(1)}} \geq \frac{y_{\sigma(2)}}{x_{\sigma(2)}} \geq \ldots \geq \frac{y_{\sigma(N)}}{x_{\sigma(N)}},$$

and the desired result follows by an easy application of Corollary 1 to the elements $\sigma(\boldsymbol{x})$ and $\sigma(\boldsymbol{y})$. ■

5. COMPARING INPUT AND OUTPUT

From now on the comparison $\boldsymbol{p}^\star \prec \boldsymbol{p}$ formalizes the notion that the output has less locality of reference than the input as this comparison captures the fact that the input pmf $\boldsymbol{p}$ is more skewed than the output pmf $\boldsymbol{p}^\star$.

5.1 A basic result

We establish comparisons between input and output popularity distributions.

THEOREM 3. *Assume the existence of the limits (7).*
(i) If $m(i;\boldsymbol{p}) \leq m(j;\boldsymbol{p})$ whenever $p(i) \leq p(j)$ for distinct $i, j = 1, \ldots, N$, then it holds that $\boldsymbol{p} \prec \boldsymbol{p}^\star$;
(ii) If $m(i;\boldsymbol{p}) \geq m(j;\boldsymbol{p})$ whenever $p(i)m(i;\boldsymbol{p}) \leq p(j)m(j;\boldsymbol{p})$ for distinct $i, j = 1, \ldots, N$, then it holds that $\boldsymbol{p}^\star \prec \boldsymbol{p}$ provided $m(i;\boldsymbol{p}) > 0$ for each $i = 1, \ldots, N$.

Proof. Under the enforced assumptions, both claims are simple consequences of Lemma 1: For Claim (i), we use $\boldsymbol{x} = \boldsymbol{p}$ and $\boldsymbol{y}$ given by $y_i = p(i)m(i;\boldsymbol{p})$, $i = 1, \ldots, N$. Note that $\bar{\boldsymbol{x}} = \boldsymbol{p}$ while $\bar{\boldsymbol{y}} = \boldsymbol{p}^\star$, and that the monotonicity assumptions hold.
For Claim (ii), we take $\boldsymbol{y} = \boldsymbol{p}$ and $\boldsymbol{x}$ given by $x_i = p(i)m(i;\boldsymbol{p})$, $i = 1, \ldots, N$. This time, we have $\bar{\boldsymbol{x}} = \boldsymbol{p}^\star$ while $\bar{\boldsymbol{y}} = \boldsymbol{p}$, and the requisite monotonicity assumptions hold. ■

Theorem 3 suggests the following definitions: We say that a caching algorithm is *bad* if its has the property that the fraction of time that a document is not in cache increases as its popularity increases, namely for every admissible pmf $\boldsymbol{p}$, it holds that $m(i;\boldsymbol{p}) \leq m(j;\boldsymbol{p})$ whenever $p(i) \leq p(j)$ for distinct $i, j = 1, \ldots, N$. For a bad caching algorithm, Claim (i) states that the output popularity pmf is more skewed than the input popularity pmf, or equivalently that the output stream displays stronger locality of reference than the input stream.
The assumptions for Claim (ii) ensure that $m(i;\boldsymbol{p}) \leq m(j;\boldsymbol{p})$ and $p(j) \leq p(i)$ occur simultaneously for distinct $i, j = 1, \ldots, N$. This leads to defining a caching algorithm as *good* if for every admissible pmf $\boldsymbol{p}$, we have $m(i;\boldsymbol{p}) \leq m(j;\boldsymbol{p})$ whenever $p(j) \leq p(i)$ for distinct $i, j = 1, \ldots, N$. Thus, a caching policy which satisfies the assumptions of Claim (ii) is necessarily a good policy. However, as we shall see shortly, this by itself is not sufficient to ensure that the output popularity pmf is more balanced than the input popularity pmf.
For any good policy, we have $m(i;\boldsymbol{p}) = m(j;\boldsymbol{p})$ whenever $p(i) = p(j)$. This observation leads to the following invariance property of good policies.

COROLLARY 2. *For any good policy, we have $\boldsymbol{p}^\star = \boldsymbol{p}$ if $\boldsymbol{p} = \boldsymbol{u}$ where $\boldsymbol{u}$ is the uniform pmf on $\{1, \ldots, N\}$.*

In other words, under good policies, the uniform pmf is a fixed-point for the input-output transformation.

5.2 A useful comparison

The following is a consequence of Lemma 1, and will be useful in the sequel.

THEOREM 4. *Let $\boldsymbol{p}$ be a pmf on $\{1, \ldots, N\}$, and for each $i = 1, \ldots, N$, define an $(N-1)$-dimensional vector*

$$\boldsymbol{p}^{(i)} := (p(1), \ldots, p(i-1), p(i+1), \ldots, p(N)).$$

For each $M = 1, 2, \ldots, N-1$, the pmf $\boldsymbol{p}^\star$ on $\{1, \ldots, N\}$ defined by

$$p^\star(i) = \frac{p(i) E_{M,N-1}(\boldsymbol{p}^{(i)})}{\sum_{j=1}^N p(j) E_{M,N-1}(\boldsymbol{p}^{(j)})} \quad i = 1, \ldots, N \tag{22}$$

satisfies the comparison $\boldsymbol{p}^\star \prec \boldsymbol{p}$.

Proof. Fix distinct $i, j = 1, \ldots, N$ and define the $(N-2)$-dimensional vector $\boldsymbol{p}^{(ij)}$ obtained from the pmf $\boldsymbol{p}$ by deleting the components associated with documents i and j. With this notation, we find

$$\begin{aligned} & E_{M,N-1}(\boldsymbol{p}^{(i)}) - E_{M,N-1}(\boldsymbol{p}^{(j)}) \\ = & \sum_{s \in \Lambda_i(M;\mathcal{N})} p(i_1) \cdots p(i_M) - \sum_{s \in \Lambda_j(M;\mathcal{N})} p(i_1) \cdots p(i_M) \\ = & \sum_{s \in \Lambda_i(M;\mathcal{N}):\ j \in s} p(i_1) \cdots p(i_M) \\ & - \sum_{s \in \Lambda_j(M;\mathcal{N}):\ i \in s} p(i_1) \cdots p(i_M) \\ = & \left(p(j) - p(i)\right) E_{M-1,N-2}(\boldsymbol{p}^{(ij)}). \end{aligned} \tag{23}$$

On the other hand, we also have

$$\begin{aligned} & p(i) E_{M,N-1}(\boldsymbol{p}^{(i)}) - p(j) E_{M,N-1}(\boldsymbol{p}^{(j)}) \\ = & \ p(i) \left(\sum_{s \in \Lambda_i(M;\mathcal{N})} p(i_1) \cdots p(i_M) \right) \\ & - p(j) \left(\sum_{s \in \Lambda_j(M;\mathcal{N})} p(i_1) \cdots p(i_M) \right) \\ = & \ p(i) \left(\sum_{s \in \Lambda_i(M;N):\ j \notin s} p(i_1) \cdots p(i_M) \right) \\ & - p(j) \left(\sum_{s \in \Lambda_j(M;N):\ i \notin s} p(i_1) \cdots p(i_M) \right) \\ = & \ \left(p(i) - p(j)\right) E_{M,N-2}(\boldsymbol{p}^{(ij)}). \end{aligned} \tag{24}$$

As we have in mind to apply Lemma 1, we take $\boldsymbol{y} = \boldsymbol{p}$ and $\boldsymbol{x}$ given by $x_i = p(i) E_{M,N-1}(\boldsymbol{p}^{(i)})$, $i = 1, \ldots, N$, whence $\bar{\boldsymbol{x}} = \boldsymbol{p}^\star$ and $\bar{\boldsymbol{y}} = \boldsymbol{p}$. For distinct $i, j = 1, \ldots, N$, we find from (23) and (24) that

$$\frac{x_i}{y_i} - \frac{x_j}{y_j} = \left(p(j) - p(i)\right) E_{M-1,N-2}(\boldsymbol{p}^{(ij)}) \leq 0$$

whenever

$$x_i - x_j = \left(p(i) - p(j)\right) E_{M,N-2}(\boldsymbol{p}^{(ij)}) \geq 0.$$

The assumptions of Lemma 1 are satisfied and the comparison $\boldsymbol{p}^\star \prec \boldsymbol{p}$ follows. ■

6. THE RANDOM POLICY

Under the random policy, when the cache is full, the document to be evicted from the cache is selected randomly according to the uniform distribution. The cache states $\{S_t,\ t = 0, 1, \ldots\}$ form a stationary ergodic Markov chain over the finite state space $\Lambda(M;\mathcal{N})$ [2, Thm. 11, p. 132] with invariant distribution given by

$$\lim_{t \to \infty} \mathbf{P}\left[S_t = s\right] = E_{M,N}(\boldsymbol{p})^{-1} p(i_1) \cdots p(i_M)$$

for every $s = \{i_1, \ldots, i_M\}$ in $\Lambda(M;\mathcal{N})$ with normalizing constant $E_{M,N}(\boldsymbol{p})$ defined at (5). Invoking ergodicity, we get the existence of the limits (7) with

$$\pi(s;\boldsymbol{p}) = E_{M,N}(\boldsymbol{p})^{-1} p(i_1) \cdots p(i_M) \tag{25}$$

for every $s = \{i_1, \ldots, i_M\}$ in $\Lambda(M;\mathcal{N})$.

As we report (25) into (10) we readily conclude that

$$\begin{aligned} m(i;\boldsymbol{p}) &= E_{M,N}(\boldsymbol{p})^{-1} \sum_{s \in \Lambda_i(M;\mathcal{N})} p(i_1) \cdots p(i_M) \\ &= \frac{E_{M,N-1}(\boldsymbol{p}^{(i)})}{E_{M,N}(\boldsymbol{p})}, \quad i = 1, \ldots, N \end{aligned} \tag{26}$$

and (8) yields the output popularity distribution as (22). The main result of this section is now immediate from Theorem 4.

THEOREM 5. *Under the random policy, it holds that $\boldsymbol{p}^\star \prec \boldsymbol{p}$.*

By going back to the proof of Theorem 4, the reader will readily check that the random policy is indeed a good policy.

In the special case $M = 1$, any demand-driven policy reduces to the policy that evicts the only document in cache if the requested document is not in cache. Specializing the results for the random policy, we find that the output pmf is given by

$$p^\star(i) = \frac{p(i)(1 - p(i))}{\sum_{j=1}^N p(j)(1 - p(j))}, \quad i = 1, \ldots, N \tag{27}$$

and Theorem 5 immediately lead to

COROLLARY 3. *With $M = 1$, under any demand-driven replacement policy, the popularity pmf $\boldsymbol{p}^\star$ of the output is given by (27), and satisfies $\boldsymbol{p}^\star \prec \boldsymbol{p}$.*

7. THE POLICY A_σ

Let σ denote a permutation of $\{1, \ldots, N\}$ which is held fixed throughout this section.

7.1 Defining the policy A_σ

Such a permutation can be used to induce an ordering of the documents by considering that the documents $\sigma(1), \sigma(2), \ldots, \sigma(N)$ are "ordered" in decreasing order. When at time $t = 0, 1, \ldots$, the cache S_t is full and the requested document R_t is not in the cache, the policy A_σ associated with the permutation σ prescribes the eviction of U_t given by

$$U_t = \arg\min\left(\sigma(j) :\ \sigma(j) \in S_t\right).$$

A well-known instance of the policy A_σ is the policy A_0 associated with the permutation $\sigma^\star$ of $\{1, \ldots, N\}$ which orders the components of the underlying pmf $\boldsymbol{p}$ in decreasing order, namely $p(\sigma^\star(1)) \geq p(\sigma^\star(2)) \geq \ldots \geq p(\sigma^\star(N))$. The eviction rule of the policy A_0 prescribes $U_t = \arg\min\left(p(j) :\ j \in S_t\right)$. It is known [2, 9] that the policy A_0 minimizes the miss rate of the cache among a large class of demand-driven policies when the input is assumed to be IRM with pmf $\boldsymbol{p}$.

7.2 Cache steady state under the policy A_σ

Under (3), every document is eventually requested with probability one, so that for sufficiently large time t the cache S_t under the replacement policy A_σ is of the form

$$S_t := \Sigma + Y_t^\sigma \tag{28}$$

with

$$\Sigma := \{\sigma(1), \sigma(2), \ldots, \sigma(M-1)\} \tag{29}$$

and

$$Y_t^\sigma \in \Sigma^c = \{\sigma(M), \ldots, \sigma(N)\}. \tag{30}$$

As explained earlier, there is then no loss of generality in assuming that the cache is indeed of the form (28)-(30), in which case the cache state S_t is determined completely by Y_t^σ. Under IRM, the rvs $\{Y_t^\sigma, t = 0, 1, \ldots\}$ form a stationary ergodic Markov chain over the finite state space Σ^c with stationary distribution $\{\pi_\sigma(y), y \in \Sigma^c\}$ described in the following lemma.

LEMMA 2. *The limits*

$$\lim_{t\to\infty} \mathbf{P}\left[Y_t^\sigma = y, R_t = x\right] = \pi_\sigma(y)p(x), \quad (x, y) \in \mathcal{N} \times \Sigma^c$$

exist, with

$$\pi_\sigma(y) = \lim_{t\to\infty} \mathbf{P}\left[Y_t^\sigma = y\right], = \frac{p(y)}{\sum_{x\notin\Sigma} p(x)}, \quad y \notin \Sigma. \tag{31}$$

The proof of Lemma 2 is omitted as it mimics the derivation of a similar result for the policy A_0 [9, Thm. 6.3, p. 268]. Note that (31) defines a pmf $\boldsymbol{\pi}_\sigma$ on Σ^c, which is simply the *conditional* pmf induced on Σ^c by the pmf $\boldsymbol{p}$.

7.3 Output popularity under the policy A_σ

From the expression of $\{\pi_\sigma(y), y \in \Sigma^c\}$ provided in Lemma 2, we obtain

$$m_\sigma(i; \boldsymbol{p}) = \begin{cases} 0 & \text{if } i \in \Sigma \\ 1 - \pi_\sigma(i) & \text{if } i \notin \Sigma \end{cases}$$

and Theorem 1 yields the output popularity distribution $\boldsymbol{p}_\sigma^\star$ as

$$p_\sigma^\star(i) = \begin{cases} 0 & \text{if } i \in \Sigma \\ \frac{p(i)(1-\pi_\sigma(i))}{\sum_{j\notin\Sigma} p(j)(1-\pi_\sigma(j))} & \text{if } i \notin \Sigma. \end{cases} \tag{32}$$

Since $p_\sigma^\star(i) = 0$ whenever i belongs to Σ, it is more natural to seek a comparison between $\boldsymbol{p}_\sigma^\star$ and the conditional pmf $\boldsymbol{\pi}_\sigma$.

THEOREM 6. *Under the policy A_σ, it holds that $\boldsymbol{p}_\sigma^\star \prec \boldsymbol{\pi}_\sigma$.*

Proof. We rewrite $\boldsymbol{p}_\sigma^\star$ in (32) as a function of $\boldsymbol{\pi}_\sigma$ by dividing its numerator and denominator by $\sum_{j\notin\Sigma} p(j)$. This yields

$$p_\sigma^\star(i) = \frac{\pi_\sigma(i)(1-\pi_\sigma(i))}{\sum_{j\notin\Sigma} \pi_\sigma(j)(1-\pi_\sigma(j))}, \quad i \notin \Sigma.$$

With Lemma 1 in mind, we take $\boldsymbol{x}$ and $\boldsymbol{y}$ to be the elements of $\mathbf{R}^{N-M+1}$ given by $\boldsymbol{y} = \boldsymbol{\pi}_\sigma$ and $x_i = \pi_\sigma(i)(1-\pi_\sigma(i))$, $i \notin \Sigma$, in which case

$$\frac{y_i}{x_i} = (1 - \pi_\sigma(i))^{-1}, \quad i \notin \Sigma. \tag{33}$$

Pick distinct i and j not in Σ. From (33), we see that $\frac{y_i}{x_i} \geq \frac{y_j}{x_j}$ if and only if $\pi_\sigma(i) \geq \pi_\sigma(j)$, and the assumptions of Lemma 1 will hold if we can show that $x_i \geq x_j$ whenever $\pi_\sigma(i) \geq \pi_\sigma(j)$. The analysis proceeds along two cases:

Case (a) – Assume $\pi_\sigma(i) \leq 1/2$. With $1/2 \geq \pi_\sigma(i) \geq \pi_\sigma(j)$, we find

$$x_i = \pi_\sigma(i)(1-\pi_\sigma(i)) \geq \pi_\sigma(j)(1-\pi_\sigma(j)) = x_j$$

by the increasing monotonicity of the mapping $p \to p(1-p)$ on the interval $[0, \frac{1}{2}]$.

Case (b) – Assume $\pi_\sigma(i) > 1/2$, in which case $1/2 > 1 - \pi_\sigma(i) \geq \pi_\sigma(j)$ since $\sum_{k\notin\Sigma} \pi_\sigma(k) = 1$. Therefore, upon applying the argument in Case (a) with $1-\pi_\sigma(i)$ and $\pi_\sigma(j)$, we still arrive at the conclusion $x_i \geq x_j$.

The assumptions of Lemma 1 are satisfied and we get the desired result with $\bar{\boldsymbol{x}} = \boldsymbol{p}_\sigma^\star$ and $\bar{\boldsymbol{y}} = \boldsymbol{\pi}_\sigma$. ■

Corollary 3 is also obtained from Theorem 6 (with $M = 1$) as expected.

8. ZIPF-LIKE PMFS

It has been observed in a number of studies that the popularity distribution of objects in request streams at Web caches is highly skewed. In [1] a good fit was provided by the *Zipf* distribution according to which the popularity of the i^{th} most popular object is inversely proportional to its rank, namely $1/i$.

In more recent studies [5, 16], "Zipf-like" distributions[7] were found more appropriate; see [5] (and references therein) for an excellent summary. Such distributions form a one-parameter family. In our set-up, for $\alpha > 0$, we say that the popularity distribution $\boldsymbol{p}$ of the $\mathcal{N}$-valued rvs $\{R_t,\ t = 0, 1, \ldots\}$ is Zipf-like with parameter α if

$$p(i) = \frac{i^{-\alpha}}{C_\alpha(N)}, \quad i = 1, \ldots, N \tag{34}$$

with

$$C_\alpha(N) := \sum_{i=1}^{N} i^{-\alpha}. \tag{35}$$

The pmf (34) will be denoted by $\boldsymbol{p}_\alpha$. It is always the case that

$$p_\alpha(1) > p_\alpha(2) > \ldots > p_\alpha(N). \tag{36}$$

Note that the case $\alpha = 1$ corresponds to the standard Zipf distribution and the value of α was typically found to be in the range $0.64 - 0.83$ [5].

Zipf-like pmfs are skewed towards the most popular objects. As $\alpha \to 0$, the Zipf-like pmf approaches the uniform distribution $\boldsymbol{u}$ while as $\alpha \to \infty$, it degenerates to the pmf $(1, 0, \ldots, 0)$. Extrapolating between these extreme cases, we expect the parameter α of Zipf-like pmfs (34)-(35) to measure the strength of skewness, with the larger α, the more skewed the pmf $\boldsymbol{p}_\alpha$. The next result shows that majorization indeed captures this fact, and so it is warranted to call α the *skewness parameter* of the Zipf-like pmf.

LEMMA 3. *For $0 < \alpha < \beta$, it holds that $\boldsymbol{p}_\alpha \prec \boldsymbol{p}_\beta$.*

Lemma 3 can already be found in [20, B.2.b, p. 130] and is an easy by-product of Lemma 1. Zipf-like pmfs will be used in the discussion of the LRU and CLIMB policies in the next two sections. Without further mention, let $\boldsymbol{p}_\alpha^\star$ denote the popularity pmf of the output induced by an input with Zipf-like popularity pmf $\boldsymbol{p}_\alpha$.

9. THE LRU POLICY

The LRU (Least-Recently-Used) policy evicts the document which was requested the least recently at the time the replacement is required.

[7] Such distributions are sometimes called generalized Zipf distributions.

9.1 LRU is a good policy

Under the IRM input, it is well known [2, Thm. 9, p. 130] [9, Thm. 6.5, p. 272] that the LRU cache states $\{\Omega_t, t = 0, 1, \ldots\}$ form a stationary ergodic Markov chain over the finite state space $\Lambda^\star(M, \mathcal{N})$ with stationary distribution given by

$$\pi^\star(s; \boldsymbol{p}) = \frac{p(i_1)\cdots p(i_M)}{\prod_{k=1}^{M-1}(1 - \sum_{j=1}^{k} p(i_j))} \quad (37)$$

for every $s = (i_1, \ldots, i_M)$ in $\Lambda^\star(M; \mathcal{N})$. It is then a simple matter to check for each $i = 1, \ldots, N$, that

$$\begin{aligned} m(i; \boldsymbol{p}) &= \sum_{s \in \Lambda_i^\star(M;\mathcal{N})} \pi^\star(s; \boldsymbol{p}) \\ &= \sum_{s \in \Lambda_i^\star(M;\mathcal{N})} \frac{p(i_1)\cdots p(i_M)}{\prod_{k=1}^{M-1}(1 - \sum_{j=1}^{k} p(i_j))}, \end{aligned} \quad (38)$$

and Theorem 1 gives the output popularity distribution in the form

$$p^\star(i) = \frac{p(i)}{K_{LRU}} \sum_{s \in \Lambda_i^\star(M;\mathcal{N})} \frac{p(i_1)\cdots p(i_M)}{\prod_{k=1}^{M-1}(1 - \sum_{j=1}^{k} p(i_j))} \quad (39)$$

where we have set $K_{LRU} := \sum_{j=1}^{N} p(j) m(j; \boldsymbol{p})$.

LEMMA 4. *The LRU policy is a good policy.*

Proof. Pick distinct $i, j = 1, \ldots, N$ with $p(j) \leq p(i)$. We need to show that

$$m(i; \boldsymbol{p}) \leq m(j; \boldsymbol{p}). \quad (40)$$

We begin by writing $m(i; \boldsymbol{p})$ as

$$\begin{aligned} m(i; \boldsymbol{p}) &= \sum_{s \in \Lambda_i^\star(M;\mathcal{N}):\ j \in s} \pi^\star(s; \boldsymbol{p}) \\ &\quad + \sum_{s \in \Lambda_i^\star(M;\mathcal{N}):\ j \notin s} \pi^\star(s; \boldsymbol{p}) \end{aligned} \quad (41)$$

with a similar expression for $m(j; \boldsymbol{p})$. Given that the sets $\{s \in \Lambda_i^\star(M; \mathcal{N}) : \ j \notin s\}$ and $\{s \in \Lambda_j^\star(M; \mathcal{N}) : \ i \notin s\}$ coincide, we find that

$$\begin{aligned} m(i; \boldsymbol{p}) - m(j; \boldsymbol{p}) &= \sum_{s \in \Lambda_i^\star(M;\mathcal{N}):\ j \in s} \pi^\star(s; \boldsymbol{p}) \\ &\quad - \sum_{s \in \Lambda_j^\star(M;\mathcal{N}):\ i \in s} \pi^\star(s; \boldsymbol{p}). \end{aligned} \quad (42)$$

The sets $\{s \in \Lambda_i^\star(M; \mathcal{N}) : \ j \in s\}$ and $\{s \in \Lambda_j^\star(M; \mathcal{N}) : \ i \in s\}$ have the same cardinality, and in fact can be put into *one-to-one* correspondence with each other as follows: Each element s in the former set does not contain i but contains j in exactly one position, say position k for some $k = 1, \ldots, M$, with all other positions occupied by neither i or j. Thus, with such an element s we can associate an element $T(s)$ in $\Lambda_j^\star(M; \mathcal{N})$ by substituting i for j at position k and letting all other positions unchanged. This element $T(s)$ now contains i but not j anymore, and is therefore an element of the latter set. Moreover, for such an element $T(s)$ it holds that

$$\pi^\star(s; \boldsymbol{p}) \leq \pi^\star(T(s); \boldsymbol{p}) \quad (43)$$

as a consequence of the assumption $p(j) \leq p(i)$ and of the expression (37). With these observation in mind, we find that

$$\begin{aligned} \sum_{s \in \Lambda_j^\star(M;\mathcal{N}):\ i \in s} \pi^\star(s; \boldsymbol{p}) &= \sum_{s \in \Lambda_i^\star(M;\mathcal{N}):\ j \in s} \pi^\star(T(s); \boldsymbol{p}) \\ &\geq \sum_{s \in \Lambda_i^\star(M;\mathcal{N}):\ j \in s} \pi^\star(s; \boldsymbol{p}) \end{aligned}$$

and the conclusion (40) is now immediate via (42). ■

9.2 A counterexample

In view of Corollary 3 and of Lemma 4, it is tempting to expect that the majorization comparison $\boldsymbol{p}^\star \prec \boldsymbol{p}$ also holds under the LRU policy. This is not the case as the following example demonstrates with $M = 3$, $N = 4$ under the Zipf-like popularity pmf (34)-(35) with $\alpha = 3$. With these parameters, we have computed the output popularity pmf under the LRU policy using (39). The numerical values of both input and output popularity pmfs are presented in Table 1.

i	1	2	3	4
$\boldsymbol{p}_\alpha$	0.8491	0.1061	0.0314	0.0133
$\boldsymbol{p}_\alpha^\star$ (LRU)	0.0118	0.2031	0.3853	0.3998
$\boldsymbol{p}_\alpha^\star$ (CLIMB)	0.0027	0.1386	0.4000	0.4587

Table 1: $\boldsymbol{p}_\alpha$, $\boldsymbol{p}_\alpha^\star$ under the LRU policy and the CLIMB policy when the input distribution is Zipf-like with parameter $\alpha = 3$

By the definition of majorization (17)-(18), the comparison $\boldsymbol{p}_\alpha^\star \prec \boldsymbol{p}_\alpha$ requires

$$\min_{i=1,\ldots,N} p_\alpha(i) \leq \min_{i=1,\ldots,N} p_\alpha^\star(i), \quad (44)$$

in clear contradiction with Table 1, and therefore does not hold. On the other hand, the comparison $\boldsymbol{p}_\alpha \prec \boldsymbol{p}_\alpha^\star$ is not valid either since it calls for the unmet requirement

$$\max_{i=1,\ldots,N} p_\alpha(i) \leq \max_{i=1,\ldots,N} p_\alpha^\star(i). \quad (45)$$

In short, $\boldsymbol{p}_\alpha$ and $\boldsymbol{p}_\alpha^\star$ are not comparable in the majorization ordering. This situation does not represent an isolated incident as the next theorem shows; its proof is available in Appendix A.1.

THEOREM 7. *Assume the input to have a Zipf-like popularity pmf $\boldsymbol{p}_\alpha$ for some $\alpha > 0$. If the number of documents N and the cache size M satisfy the condition*

$$N < M! \quad (46)$$

then under the LRU policy, there exists $\alpha^\star = \alpha^\star(M, N)$ such that $\boldsymbol{p}_\alpha^\star \prec \boldsymbol{p}_\alpha$ does not hold whenever $\alpha > \alpha^\star$.

9.3 A conjecture

Theorems 5 and 6 were valid for *all* values of M and N, and for *arbitrary* admissible pmfs. While the counterexamples contained in Theorem 7 dash our hopes to get an analogous result for the LRU policy, the possibility remains, fueled by Corollaries 2 and 3, that the positive result $\boldsymbol{p}^\star \prec \boldsymbol{p}$ is nevertheless valid in some appropriate range of the parameters M and N. We now explore this issue still with Zipf-like popularity pmfs (34)-(35).

CONJECTURE 1. *Assume that the popularity pmf is the Zipf-like distribution (34)-(35) with $\alpha > 0$. For each $N = 1, \ldots$, there exists an integer $M^\star = M^\star(\alpha; N)$ with $1 \leq M^\star < N$ such that $\boldsymbol{p}_\alpha^\star \prec \boldsymbol{p}_\alpha$ under the LRU policy whenever $M = 1, \ldots, M^\star$.*

In support of this conjecture, we have carried out simulations of the cache operating under the LRU policy when the input pmf is

Zipf-like with parameter $\alpha = 0.8, 1$ and 2 and the number of documents $N = 1,000$.[8] We find the output popularity pmfs for different values of cache size, namely $M = 10, 50, 100, 500$. The resulting output popularity pmfs in the original order of documents are shown in Figure 1, while the results after rearranging documents in the decreasing order of their output probabilities are displayed in Figure 2.

From Figure 2 (a), when $\alpha = 0.8$, the comparison $\boldsymbol{p}^\star_\alpha \prec \boldsymbol{p}_\alpha$ holds for $M = 10, 50$. Indeed, from their respective plots, we observe that the pmfs $\boldsymbol{p}_\alpha$ and $\boldsymbol{p}^\star_\alpha$ when arranged in decreasing order intersect only once, namely $p^\star_\alpha([i]) \leq p_\alpha(i)$, $i = 1, \ldots, k$, and $p^\star_\alpha([i]) \geq p_\alpha(i)$, $i = k+1, \ldots, N$, for some $k = 1, \ldots, N-1$, where $p^\star_\alpha([1]) \geq p^\star_\alpha([2]) \geq \ldots \geq p^\star_\alpha([N])$ are the components of $\boldsymbol{p}^\star_\alpha$ arranged in decreasing order. This is the sufficient condition for the majorization comparison provided in Proposition 1.

However, for $\alpha = 0.8$ and $M = 100, 500$, despite the fact that in Figure 2 (a), $\boldsymbol{p}^\star_\alpha$ of both cases look uniform in the range where document rank is smaller than M, the comparison $\boldsymbol{p}^\star_\alpha \prec \boldsymbol{p}_\alpha$ is invalid since the necessary condition (44) does not hold. This violation, $\min_{i=1,\ldots,N} p^\star_\alpha(i) < p_\alpha(N)$, can be easily seen from Figure 1 (a) or the subplot inside Figure 2 (a).

For $\alpha = 1$ and 2, by the same arguments, we conclude from Figure 1 (b)-(c) and 2 (b)-(c) that the comparison $\boldsymbol{p}^\star_\alpha \prec \boldsymbol{p}_\alpha$ holds for $M = 10$ but does not hold for other cache sizes $M = 50, 100, 500$. Therefore, the experimental results agree with Conjecture 1. These calculations suggest that the value of $M^\star(\alpha; N)$ in Conjecture 1 decreases as α increases.

10. THE CLIMB POLICY

The CLIMB policy ranks documents in cache according to their recency of access: If the request document is not in the cache, the document at the last position (position M) is evicted and replaced by the new document. If the requested document is in the cache at position i, $i = 2, \ldots, M$, it exchanges position with the document at position $i-1$. The cache remains unchanged if the requested document is in the cache at position 1.

10.1 CLIMB is a good policy

Under the IRM assumption on the input, the cache states $\{\Omega_t, t = 0, 1, \ldots\}$ form a stationary ergodic Markov chain on the finite state space $\Lambda^\star(M, \mathcal{N})$ and the stationary distribution is given by [2, p. 133]

$$\pi^\star(s; \boldsymbol{p}) = \frac{1}{K_{CL}} \prod_{\ell=1}^{M} p(i_\ell)^{M-\ell+1} \quad (47)$$

for each $s = (i_1, \ldots, i_M)$ in $\Lambda^\star(M; \mathcal{N})$, where the normalizing constant is simply

$$K_{CL} := \sum_{(i_1,\ldots,i_M)\in\Lambda^\star(M;\mathcal{N})} \prod_{\ell=1}^{M} p(i_\ell)^{M-\ell+1}.$$

Thus, for each $i = 1, \ldots, N$, we find

$$\begin{aligned} m(i; \boldsymbol{p}) &= \sum_{s\in\Lambda^\star_i} \pi^\star(s; \boldsymbol{p}) \\ &= \frac{1}{K_{CL}} \sum_{s\in\Lambda^\star_i(M;\mathcal{N})} \prod_{\ell=1}^{M} p(i_\ell)^{M-\ell+1} \quad (48) \end{aligned}$$

[8] We choose simulations over numerical evaluation of (39) because these expressions are not suitable for numerical evaluation due to a combinatorial explosion.

and

$$p^\star(i) = \frac{p(i)}{K'_{CL}} \sum_{s\in\Lambda^\star_i(M;\mathcal{N})} \prod_{\ell=1}^{M} p(i_\ell)^{M-\ell+1} \quad (49)$$

with $K'_{CL} := K_{CL} \sum_{j=1}^{N} p(j) m(j; \boldsymbol{p})$.

LEMMA 5. *The CLIMB policy is a good policy.*

Proof. The proof is essentially that for the analogous result for the LRU policy given in Lemma 4. Here the validity of (43) follows from the expressions (47). ■

10.2 A counterexample

Again, Corollary 3 and Lemma 5 might lead one to expect that the majorization comparison $\boldsymbol{p}^\star \prec \boldsymbol{p}$ also holds under the CLIMB policy. This is not the case as the following example demonstrates with $M = 3$, $N = 4$ and Zipf-like popularity pmf (34)-(35) with $\alpha = 3$. With these parameters, we have computed the output popularity pmf under the CLIMB policy using (49). The numerical values of both input and output popularity pmfs are presented in Table 1.

As in the case of the LRU policy, the pmfs $\boldsymbol{p}_\alpha$ and $\boldsymbol{p}^\star_\alpha$ are not comparable in the majorization ordering. The arguments are similar to the one provided for the LRU policy, and are therefore omitted. Moreover, a result analogous to Theorem 7 holds for the CLIMB policy. It is given next, with a proof available in Appendix A.2.

THEOREM 8. *Assume the input to have a Zipf-like popularity pmf $\boldsymbol{p}_\alpha$ for some $\alpha > 0$. If the number of documents N and the cache size M satisfy the condition (46), then under the CLIMB policy, there exists $\alpha^\star = \alpha^\star(M, N)$ such that $\boldsymbol{p}^\star_\alpha \prec \boldsymbol{p}_\alpha$ does not hold whenever $\alpha > \alpha^\star$.*

10.3 A conjecture

As in the case of the LRU policy, we venture that a conjecture similar to Conjecture 1 is also valid for the CLIMB policy when the input popularity pmf is a Zipf-like distribution. A number of simulation experiments have been carried out under the CLIMB policy as was done for the LRU policy; they are omitted due to space limitation. However, they again support the validity of the conjecture.

11. REFERENCES

[1] V. Almeida and A. Bestavros and M. Crovella and A. de Oliveira, "Characterizing reference locality in the Web," in Proceedings of PDIS'96, December 1996, Miami (FL), pp. 92-107.

[2] O.I. Aven and E.G. Coffman and Y.A. Kogan, *Stochastic Analysis of Computer Storage*. D. Reidel Publishing Company, Dordrecht (Holland), 1987.

[3] G. Barish and K. Obraczka, "World Wide Web caching: Trends and techniques," IEEE Communications Magazine, Internet Technology Series. May 2000.

[4] L.A. Belady, "A study of replacement algorithms for a virtual-storage computer," *IBM Systems Journal* **5** (1966), pp. 78-101.

[5] L. Breslau, P. Cao, L. Fan, G. Phillips and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in Proceedings of IEEE INFOCOM 1999, New York (NY), March 1999.

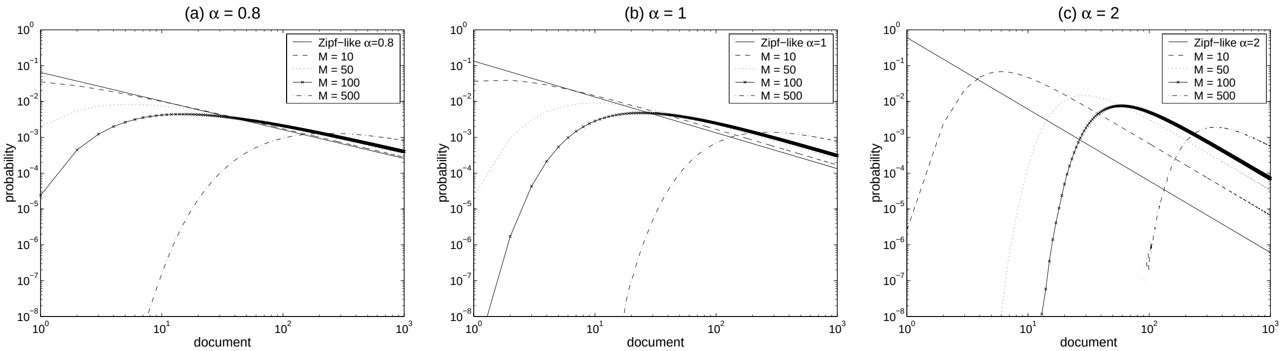

Figure 1: LRU output popularity pmf with different cache size M when the input has a Zipf-like pmf with (a) $\alpha = 0.8$, (b) $\alpha = 1$ and (c) $\alpha = 2$

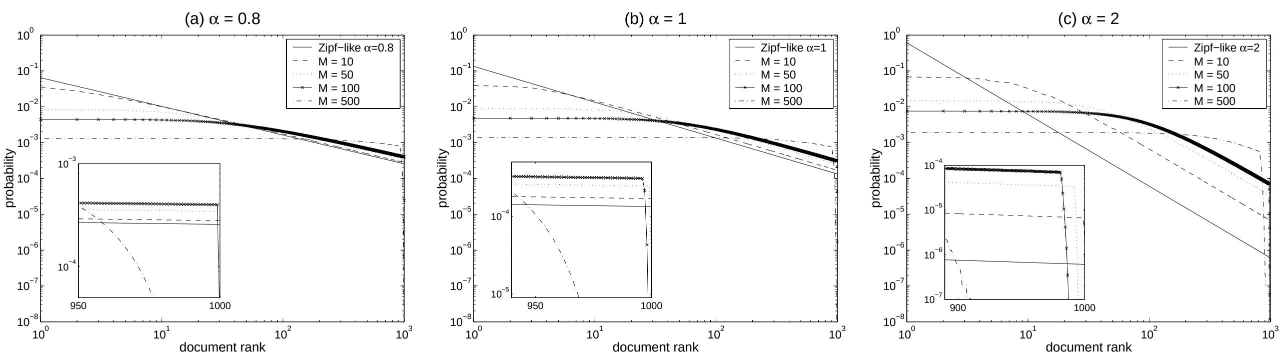

Figure 2: LRU output popularity pmf with different cache size M when the input has a Zipf-like pmf with (a) $\alpha = 0.8$, (b) $\alpha = 1$ and (c) $\alpha = 2$. Documents are ranked according to their probabilities.

[6] A. Chankhunthod, P.B. Dantzig, C. Neerdaels, E.J. Schwartz and K. Worrell, "A hierarchical Internet object cache," In the Proceedings of the USENIX 1995 Technical Conference, New Orleans (LA), 1995.

[7] H. Che, Z. Wang and Y. Tung, "Analysis and design of hierarchical web caching systems," In Proceedings of IEEE INFOCOM 2001, Anchorage (AL), April 2001.

[8] K.L. Chung, *A Course in Probability Theory*, Second. Edition, Academic Press, New York (NY), 1974.

[9] E. Coffman and P. Denning, *Operating Systems Theory*, Prentice-Hall, NJ, 1973.

[10] S. Christodoulakis, "Implications of certain assumptions in database performance evaluation," *ACM Transactions on Database Systems* **9** (1984), pp. 163-186.

[11] P.J. Denning, "The working set model for program behavior," *Communications of the ACM* **11** (1968), pp. 323-333.

[12] S.G. Dykes and K.A. Robbins, "A viability analysis of cooperative proxy caching," In Proceedings of IEEE INFOCOM 2001, Anchorage (AL), April 2001.

[13] S.G. Dykes, C.L. Jeffrey and S. Das, "Taxonomy and design analysis for distributed web caching," In Proceedings of the 32th Hawaii International Conference on System Sciences (HICSS '99), Maui (HI), January 1999.

[14] J. Escorcia, D. Dipak and D. Sarkar, "A novel cache distribution heuristic algorithm for a mesh of caches and its performance evaluation," *Computer Communications* **25** (2002), pp. 329-340.

[15] R. Fonseca, V. Almeida, M. Crovella and B. Abrahao, "On the intrinsic locality of web reference streams," IEEE INFOCOM 2003, San Francisco (CA), April 2003.

[16] S. Jin and A. Bestavros, "GreedyDual* Web caching algorithm: Exploiting the two sources of temporal locality in Web request streams," In Proceedings of the 5th International Web Caching and Content Delivery Workshop, Lisbon, Portugal, May 2000.

[17] S. Jin and A. Bestavros, "Sources and Characteristics of Web Temporal Locality," In Proceedings of MASCOTS'2000: The IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, San Francisco (CA), 2000.

[18] A. Mahanti, C. Williamson and D. Eager, "Traffic Analysis of a web proxy caching hierarchy," *IEEE Network* **14** (2000), May/June 2000, pp. 16-23.

[19] A. Mahanti, C. Williamson and D. Eager, "Temporal locality and its impact on Web proxy cache performance," *Performance Evaluation* **42** (2000), Special Issue on Internet Performance Modeling, pp. 187-203.

[20] A.W. Marshall and I. Olkin, *Inequalities: Theory of Majorization and Its Applications*, Academic Press, New York (NY), 1979.

[21] S. Vanichpun and A.M. Makowski, "Comparing strength of locality of reference – Popularity, majorization, and some folk theorems." To appear in IEEE INFOCOM 2004.

[22] J. Wang, "A survey of web caching schemes for the Internet," Technical Report TR99-1747, Department of Computer Science, Cornell University, Ithaca (NY), May 1999.

[23] C. Williamson, "On filter effects in caching hierarchies," *ACM Transactions on Internet Technology* **2** (2002), pp. 47-77.

[24] A. Wolman, G.M. Voelker, N. Sharma, N. Cardwell, A. Karlin and H.M. Levy, "On the scale and performance of cooperative Web proxy caching," *Operating Systems Review* **35** (1999), pp. 16-31.

[25] P.C. Yue and C.K. Wong "On the optimality of the probability ranking scheme in storage applications," *Journal of the ACM* **20** (1973), pp. 624-633.

APPENDIX

The proofs of Theorem 7 and of Theorem 8 proceeds along similar lines: By the definition of majorization (17)-(18), the comparison $\boldsymbol{p}^\star_\alpha \prec \boldsymbol{p}_\alpha$ requires the condition (44) to hold. Thus, as we recall (36), this comparison will not hold if we can show that

$$C_\alpha(N)N^\alpha \cdot \min_{i=1,\ldots,N} p^\star_\alpha(i) < 1. \tag{50}$$

In each case, we show under the appropriate conditions on M and N that (50) indeed holds for large enough values of α. To do so, we shall have repeated use for the next elementary lemma where asymptotic equivalence is defined as follows: For mappings $f, g : \mathbf{R}_+ \to \mathbf{R}$, we write $f(\alpha) \sim g(\alpha)$ $(\alpha \to \infty)$ if $\lim_{\alpha\to\infty} \frac{f(\alpha)}{g(\alpha)} = 1$.

LEMMA 6. *Consider a finite family $a_1, \ldots, a_K$ of positive scalars. We have*

$$\sum_{k=1} a_k^{-\alpha} \sim c \cdot \left(\min_{k=1,\ldots,K} a_k \right)^{-\alpha} \quad (\alpha \to \infty).$$

where c denotes the number of indices ℓ for which it holds $a_\ell = \min_{k=1,\ldots,K} a_k$.

In what follows, without further mention, all asymptotics are understood in the regime where α is large, and the qualifier $\alpha \to \infty$ is now dropped from the notation. In particular, we have $C_\alpha(N) \sim 1$.

A.1 A Proof of Theorem 7

Fix $\alpha > 0$ and substitute (34)-(35) into the expressions (37)-(39) for the pmf $\boldsymbol{p}^\star_\alpha$. For each $i = 1, \ldots, N$, we find

$$p^\star_\alpha(i) = \frac{i^{-\alpha}\nu_\alpha(i)}{\sum_{j=1}^N j^{-\alpha}\nu_\alpha(j)} \tag{51}$$

with

$$\nu_\alpha(i) = \frac{1}{C_\alpha(N)} \sum_{s \in \Lambda^\star_i(M;\mathcal{N})} \frac{\prod_{\ell=1}^M i_\ell^{-\alpha}}{\prod_{k=1}^{M-1} \left(\sum_{j \notin \{i_1,\ldots,i_k\}} j^{-\alpha} \right)}. \tag{52}$$

where we have written $j \notin \{i_1, \ldots, i_k\}$ to denote the set of elements j in $\mathcal{N}$ which are not in the set $\{i_1, \ldots, i_k\}$.

Fix $i = 1, 2, \ldots, N$. For any element $s = (i_1, \ldots, i_M)$ in $\Lambda^\star_i(M;\mathcal{N})$, we invoke Lemma 6 to claim that

$$\sum_{j \notin \{i_1,\ldots,i_k\}} j^{-\alpha} \sim \left(\min_{j \notin \{i_1,\ldots,i_k\}} j \right)^{-\alpha} \tag{53}$$

for each $k = 1, \ldots, M-1$, whence

$$\prod_{k=1}^{M-1} \left(\sum_{j \notin \{i_1,\ldots,i_k\}} j^{-\alpha} \right) \sim \rho(s)^{-\alpha} \tag{54}$$

where we have set $\rho(s) := \prod_{k=1}^{M-1} \left(\min_{j \notin \{i_1,\ldots,i_k\}} j \right)$. An additional use of Lemma 6 leads readily to

$$\nu_\alpha(i) \sim \sum_{s \in \Lambda^\star_i(M;\mathcal{N})} \left(\frac{\prod_{\ell=1}^M i_\ell}{\rho(s)} \right)^{-\alpha} \sim c(i) \cdot \nu(i)^{-\alpha} \tag{55}$$

where

$$\nu(i) = \min_{s \in \Lambda^\star_i(M;\mathcal{N})} \left(\frac{\prod_{\ell=1}^M i_\ell}{\rho(s)} \right) \tag{56}$$

and $c(i)$ is the number of elements s in $\Lambda^\star_i$ that achieve the minimum in (56).

It is clear that

$$\nu(i) \geq \frac{\min_{s \in \Lambda^\star_i(M;\mathcal{N})} \left(\prod_{\ell=1}^M i_\ell \right)}{\max_{s \in \Lambda^\star_i(M;\mathcal{N})} \rho(s)} \tag{57}$$

and we will show that there exists s in $\Lambda^\star_i(M;\mathcal{N})$ such that (57) holds as an equality, thereby providing the minimal value of $\nu(i)$. For $i = M+1, \ldots, N$, it is plain that $\prod_{\ell=1}^M i_\ell$ attains its minimum at $M!$ while $\rho(s)$ also attains its maximum at $M!$ with $s = (1, \ldots, M)$ being the only element in $\Lambda^\star_i$ which achieves both extreme values. This claim can be realized easily by basic interchange arguments. Thus, $c(i) = 1$ and

$$\nu(i) = \frac{M!}{M!} = 1. \tag{58}$$

Similarly, when $i = 1, \ldots, M$, the element $s = (1, \ldots, i-1, i+1, \ldots, M+1)$ gives the minimum of $\prod_{\ell=1}^M i_\ell$ at the value $\prod_{\ell=1}^{i-1} \ell \cdot \prod_{\ell=i+1}^{M+1} \ell$ and the maximum of $\rho(s)$ at $\prod_{\ell=2}^{i-1} \ell \cdot i^{M-i+1}$, whence

$$\nu(i) = \frac{\prod_{\ell=1}^{i-1} \ell \cdot \prod_{\ell=i+1}^{M+1} \ell}{\prod_{\ell=2}^{i-1} \ell \cdot i^{M-i+1}} = \frac{(M+1)!}{i! i^{M-i+1}}. \tag{59}$$

Note that $c(i) = 1$ for $i = 2, \ldots, M$ but $c(1) = M!$ because when $i = 1$, $\rho(s) = 1$ for *any* element s in $\Lambda^\star_1$ and any element s in $\Lambda^\star_1$ containing $\{2, \ldots, M+1\}$ attains the minimal value (59).

Invoking Lemma 6 again, we find

$$\sum_{i=1}^N i^{-\alpha}\nu_\alpha(i) \sim c \cdot \left(\min_{i=1,\ldots,N} i\nu(i) \right)^{-\alpha} \tag{60}$$

where c denoting the number of indices achieving the minimum in $\min_{i=1,\ldots,N} i\nu(i)$. It follows from (58) that

$$\min_{i=M+1,\ldots,N} i\nu(i) = \min_{i=M+1,\ldots,N} i = M+1 \tag{61}$$

and (59) allows us to write

$$\min_{i=1,\ldots,M} i\nu(i) = (M+1) \min_{i=1,\ldots,M} \varphi(i). \tag{62}$$

with

$$\varphi(i) := \frac{M!}{i! i^{M-i}}, \quad i = 1, \ldots, M. \tag{63}$$

It is a simple matter to check that

$$M! = \varphi(1) > \ldots > \varphi(M) = 1, \tag{64}$$

so that the minimum in (62) is achieved at $i = M$ with value $\min_{i=1,\ldots,M} i\nu(i) = M+1$. It then follows from this fact and (61) that

$$\min_{i=1,\ldots,N} i\nu(i) = M+1 \tag{65}$$

and $c = 2$. By virtue of (51), (55) (60) and (65), we can now write

$$p^\star_\alpha(i) \sim \frac{c(i)}{2}\left(\frac{M+1}{i\nu(i)}\right)^\alpha.$$

Consequently,

$$\min_{i=1,\ldots,N} p^\star_\alpha(i) \quad\sim\quad \frac{1}{2}\min_{i=1,\ldots,N}\left(c(i)\left(\frac{M+1}{i\nu(i)}\right)^\alpha\right). \tag{66}$$

Again, by recalling (58), we get

$$\min_{i=M+1,\ldots,N}\left(c(i)\left(\frac{M+1}{i\nu(i)}\right)^\alpha\right) = \left(\frac{M+1}{N}\right)^\alpha \tag{67}$$

where the minimum is achieved at $i = N$. On the other hand, by using (59), we get with the help of (63) that

$$\min_{i=2,\ldots,N}\left(c(i)\left(\frac{M+1}{i\nu(i)}\right)^\alpha\right) = \left(\frac{2^{M-1}}{M!}\right)^\alpha \tag{68}$$

where the minimum is achieved at $i = 2$. Finally, $\nu(1) = (M+1)!$ and

$$c(1)\left(\frac{M+1}{\nu(1)}\right)^\alpha = M!\frac{1}{(M!)^\alpha} \tag{69}$$

Combining (67), (68) and (69), we conclude from (66) that

$$\begin{aligned} & C_\alpha(N)N^\alpha \cdot \min_{i=1,\ldots,N} p^\star_\alpha(i) \\ \sim\ & \frac{1}{2}\min\left(M!\left(\frac{N}{M!}\right)^\alpha, \left(\frac{2^{M-1}N}{M!}\right)^\alpha, (M+1)^\alpha\right). \end{aligned}$$

Under (46), as α grows large, the first term in the minimum above will have the smallest value, so

$$C_\alpha(N)N^\alpha \cdot \min_{i=1,\ldots,N} p^\star_\alpha(i) \sim \frac{M!}{2}\left(\frac{N}{M!}\right)^\alpha,$$

and for large enough values of α, the condition (50) indeed holds. ■

A.2 A Proof of Theorem 8

Fix $\alpha > 0$. Substituting (34)-(35) into the expressions (47)-(49), we find for each $i = 1, \ldots, N$, that

$$p^\star_\alpha(i) = \frac{i^{-\alpha}\mu_\alpha(i)}{\sum_{j=1}^N j^{-\alpha}\mu_\alpha(j)} \tag{70}$$

with

$$\mu_\alpha(i) = \sum_{s\in\Lambda^\star_i(M;\mathcal{N})}\prod_{\ell=1}^M i_\ell^{-\alpha(M-\ell+1)}. \tag{71}$$

Fix $i = 1, \ldots, N$. We immediately get

$$\mu_\alpha(i) \sim \mu(i)^{-\alpha} \tag{72}$$

with

$$\mu(i) := \min_{s\in\Lambda^\star_i(M;\mathcal{N})}\left(\prod_{\ell=1}^M i_\ell^{M-\ell+1}\right). \tag{73}$$

Elementary interchange arguments show that the minimal value in (73) is achieved at some unique element $s = (i_1, \ldots, i_M)$ of $\Lambda^\star_i(M;\mathcal{N})$ with the property $i_1 < i_2 < \ldots < i_M$.

Using this observation, we first conclude that

$$\mu(M+1) = \ldots = \mu(N) = \prod_{\ell=1}^M \ell^{M-\ell+1}. \tag{74}$$

On the other hand, whenever $i = 1, 2, \ldots, M$, direct inspection shows that

$$\begin{aligned} \mu(i) &= (M+1)\prod_{1\le\ell<i}\ell^{M-\ell+1}\cdot\prod_{i<\ell\le M}\ell^{M-\ell+2} \\ &= \frac{\prod_{i<\ell\le M}\ell}{i^{M-i+1}}\cdot(M+1)\mu(M+1) \\ &= (M+1)\mu(M+1)\frac{\varphi(i)}{i} \end{aligned} \tag{75}$$

where $\varphi(i), i = 1, \ldots, M$, are defined in (63).

Next, upon making use of Lemma 6 again, we see that

$$\sum_{i=1}^N i^{-\alpha}\mu_\alpha(i) \sim c\cdot\left(\min_{i=1,\ldots,N} i\mu(i)\right)^{-\alpha} \tag{76}$$

with c denoting the number of indices achieving the minimum in $\min_{i=1,\ldots,N} i\mu(i)$. Obviously, by virtue of (74), we find

$$\min_{i=M+1,\ldots,N} i\mu(i) = (M+1)\mu(M+1) \tag{77}$$

where the minimum is achieved at $i = M+1$. On the other hand, as we rely on (75), we see that

$$\min_{i=1,\ldots,M} i\mu(i) = (M+1)\mu(M+1)\min_{i=1,\ldots,M}\varphi(i) \tag{78}$$

and by noting (64), the minimum in (78) is achieved at $i = M$ with value $\min_{i=1,\ldots,M} i\mu(i) = (M+1)\mu(M+1)$. Combining this fact and (77) leads to

$$\min_{i=1,\ldots,N} i\mu(i) = (M+1)\mu(M+1) \tag{79}$$

with $c = 2$. It is now plain to see from (70), (72),(76) and (79) that

$$p^\star_\alpha(i) \quad\sim\quad \frac{1}{2}\left(\frac{(M+1)\mu(M+1)}{i\mu(i)}\right)^\alpha. \tag{80}$$

From (80) we conclude that

$$\min_{i=1,\ldots,N} p^\star_\alpha(i) \sim \frac{1}{2}\left(\frac{(M+1)\mu(M+1)}{\max_{i=1,\ldots,N} i\mu(i)}\right)^\alpha. \tag{81}$$

where

$$\max_{i=1,\ldots,N} i\mu(i) = \max\left((M+1)!, N\right)\cdot\mu(M+1). \tag{82}$$

This last relation can be derived by applying to (75) an analysis similar to the one used for validating (79). To conclude the proof, we note from (81) and (82) that

$$C_\alpha(N)N^\alpha\cdot\min_{i=1,\ldots,N} p^\star_\alpha(i) \sim \frac{1}{2}\left(\frac{(M+1)N}{\max\left((M+1)!, N\right)}\right)^\alpha$$

and that $\max\left((M+1)!, N\right) = (M+1)!$ under (46). Consequently, the last asymptotics takes the simplified form

$$C_\alpha(N)N^\alpha\cdot\min_{i=1,\ldots,N} p^\star_\alpha(i) \sim \frac{1}{2}\left(\frac{N}{M!}\right)^\alpha,$$

clearly ensuring the required validity of (50) for large enough values of α. ■