Asymptotic Optimality of the Static Frequency Caching in the Presence of Correlated Requests*

Predrag R. Jelenković[†] Ana Radovanović[‡]

Abstract

Renewed interest in caching algorithms stems from their application to content distribution on the Web. When documents are of equal size and their requests are independent and equally distributed, it is well known that static algorithm that keeps the most frequently requested documents in the cache is optimal. However, there are no explicit caching algorithms that are provably optimal when the requests are statistically correlated. In this paper, we show, maybe somewhat surprisingly, that keeping the most frequently requested documents in the cache is still optimal for large cache sizes even if the requests are strongly correlated. We model the statistical dependency of requests using semi-Markov modulated processes that can capture strong statistical correlation, including the empirically observed long-range dependence in the Web access sequences.

Although frequency algorithm and its practical version least-frequently-used policy is not commonly used in practice due to their complexity and static nature, our result provides a benchmark for evaluating the popular heuristic schemes. In particular, an important corollary of our main theorem and recent result from [9] is that the widely used least-recently-used heuristic is asymptotically near-optimal under the semi-Markov modulated requests and generalized Zipf's law document frequencies.

Keywords: Web caching, cache fault probability, average-case analysis, lest-frequently-used caching, least-recently-used caching, semi-Markov processes, long-range dependence

1 Introduction

One of important problems facing current and future network designs is the ability to store and efficiently deliver a huge amount of multimedia information in a timely manner. Web caching is widely recognized as an effective solution that improves the efficiency and scalability of multimedia content delivery, benefits of which have been repeatedly verified in practice.

Caching is essentially a process of storing information closer to users so that Internet service providers, delivering a given content, do not have to go back to the origin servers every time the content is requested. It is clear that keeping more popular documents closer to the users can significantly reduce the traffic between the cache and the main servers and, therefore, improve the network performance, i.e., reduce the download latency and network congestion. One of the key components of engineering efficient Web caching systems is designing document placement/replacement algorithms (policies) that are managing cache content, i.e., selecting and possibly dynamically updating a collection of cached documents.

The main tendency in creating and implementing these algorithms is minimizing the long-term fault probability, i.e., the average number of misses during a long time period. In the context of equal size documents and independent reference model, i.e., independent and identically distributed requests, it is well known (see [5], Chapter 6 of [13]) that keeping the most popular documents in the cache optimizes the long term cache performance; throughout this paper we refer to this algorithm as *static frequency caching*. A practical implementation of this algorithm is known as Least-Frequently-Used rule (LFU). However, the previous model does not incorporate any of the recently observed properties of the Web environment, such as: variability of document sizes, presence of temporal locality in the request patterns (e.g., see [8], [12], [2], [6], [7] and references therein), variability in document popularities (e.g., see [3]) and retrieval latency (e.g., see [1]).

Many heuristic algorithms that exploit the previously mentioned properties of the Web environment have been proposed (e.g., see [7], [5], [11] and references therein). However, there are no explicit algorithms that are provably optimal when the requests are statistically correlated even if documents are of equal size. Our main result of this paper, stated in Theorem 3.1 of Section 3, shows that, under the general assumptions of semi-Markov modulated requests, the static frequency caching algorithm is still optimal for large cache sizes. The semi-Markov modulated processes, described in Section 2, are capable of modeling a wide range of statistical correlation, including the long-range dependence (LRD) that was repeatedly experimentally observed in Web access patterns; this type of models was recently used in [9] and their potential confirmed on real Web traces in [8]. In Section 4, under mild additional assumptions, we show how our result extends to variable page sizes. Our optimality result provides a benchmark for evaluating other heuristic schemes,

*This work is supported by the NSF Grant No. 0092113.

[†] Electrical Engineering Department, Columbia University.

[‡] Mathematical Sciences Department, IBM Research.

suggesting that any heuristic caching policy that approximates well the static frequency caching should achieve the nearly-optimal performance for large cache sizes. In particular, in conjunction with our result from [9], we show that a widely implemented Least-Recently-Used (LRU) caching heuristic is, for semi-Markov modulated requests and generalized Zipf's law document frequencies, asymptotically only a factor of 1.78 away from the optimal.

2 Modeling statistical dependency in the request process

In this section we describe a semi-Markov modulated request process. As stated earlier, this model is capable of capturing a wide range of statistical correlation, including the commonly empirically observed LRD. This approach was recently used in [9], where one can find more details and examples.

Let a sequence of requests arrive at Poisson points $\{\tau_n, -\infty < n < \infty\}$ of unit rate. At each point τ_n, we use R_n, $R_n \in \{1, 2, \dots, \}$, to denote a document that has been requested, i.e., the event $\{R_n = i\}$ represents a request for document i at time τ_n; we assume that the sequence $\{R_n\}$ is independent of the arrival Poisson points $\{\tau_n\}$ and that $\mathbb{P}[R_n = i] > 0$ for all i and $\mathbb{P}[R_n < \infty] = 1$.

Next, we describe the dependency structure of the request sequence $\{R_n\}$. We consider the class of finite-state, stationary and ergodic semi-Markov processes J, with jumps at almost surely strictly increasing points $\{T_n, -\infty < n < \infty\}$, $T_0 \le 0 < T_1$. The process $\{J_{T_n}, -\infty < n < \infty\}$ is an irreducible Markov chain with finitely many states $\{1, \dots, M\}$ and transition matrix $\{p_{ij}\}$. The explicit construction of process J_t, $t \in \mathbb{R}$ is presented in Subsection 4.3 of [9]. In addition, J_t is constructed piecewise constant and right-continuous *modulating process*, where

$$J_t = J_{T_n}, \quad \text{if} \qquad T_n \le t < T_{n+1}.$$

Let $\pi_r = \mathbb{P}[J_t = r]$, $1 \le r \le M$, be the stationary distribution of J and independent of Poisson points $\{\tau_n\}$. To avoid trivialities, we assume that $\min_r \pi_r > 0$. For each $1 \le r \le M$, let $q_i^{(r)}, 1 \le i \le N \le \infty$, be a probability mass function; $q_i^{(r)}$ is used to denote the probability of requesting item i when the underlying process J is in state r. Next, the dynamics of R_n are uniquely determined by the modulating process J according to the following equation

(2.1)
$$\mathbb{P}[R_l = i_l, 1 \le l \le n | J_t, t \le \tau_n] = \prod_{l=1}^{n} q_{i_l}^{(J_{\tau_l})}, \qquad n \ge 1,$$

i.e., the sequence of requests R_n is conditionally independent given the modulating process J. Given the properties introduced above, it is easy to conclude that the constructed request process $\{R_n\}$ is stationary and ergodic as well. We will use

$$q_i = \mathbb{P}[R_n = i] = \sum_{r=1}^{M} \pi_r q_i^{(r)}$$

to express the marginal request distribution, with the assumption that $q_i > 0$ for all $i \ge 1$. In addition, assume that requests are enumerated according to the non-increasing order of marginal request popularities, i.e., $q_1 \ge q_2 \ge \dots$. The preceding processes are constructed on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$.

In this paper we are using the following standard notation. For any two real functions $a(t)$ and $b(t)$ and fixed $t_0 \in \mathbb{R} \cup \{\infty\}$ we will use $a(t) \sim b(t)$ as $t \to t_0$ to denote $\lim_{t \to t_0} [a(t)/b(t)] = 1$. Similarly, we say that $a(t) \gtrsim b(t)$ as $t \to t_0$ if $\liminf_{t \to t_0} a(t)/b(t) \ge 1$; $a(t) \lesssim b(t)$ has a complementary definition.

Throughout the paper we will exploit the renewal (regenerative) structure of the semi-Markov process. In this regard, let $\{\mathcal{T}_i\}$, $\mathcal{T}_0 \le 0 < \mathcal{T}_1$, be a subset of points $\{T_n\}$ for which $J_{T_n} = 1$. Then, it is well known that $\{\mathcal{T}_i\}$ is a renewal process and that sets of variables $\{J_t, \mathcal{T}_j \le t < \mathcal{T}_{j+1}\}$ are independent for different j and identically distributed, i.e., $\{\mathcal{T}_i\}$ are regenerative points for $\{J_t\}$. Furthermore, the conditional independence of $\{R_n\}$ given $\{J_t\}$, implies that $\{\mathcal{T}_i\}$ are regenerative points for J_n as well.

Next we define $\mathcal{R}(u, t)$, $1 \le r \le M$, to be a set of distinct requests that arrived in interval $[u, t)$, $u \le t$, and denote by $N_r(u, t)$, $1 \le r \le M$, the number of requests in interval $[u, t)$ when process J_t is in state r. Furthermore, let $N(u, t) \triangleq N_1(u, t) + \cdots + N_M(u, t)$ representing the total number of requests in $[u, t)$; note that $N(u, t)$ is Poisson with mean $t - u$.

The following technical lemma will be used in the proof of the main result of this paper.

LEMMA 2.1. *For the request process introduced above, the following asymptotic relation holds:*

(2.2) $$\mathbb{P}[i \in \mathcal{R}(\mathcal{T}_1, \mathcal{T}_2)] \sim q_i \mathbb{E}[\mathcal{T}_2 - \mathcal{T}_1] \quad \textit{as} \quad i \to \infty,$$

where $\mathcal{R}(u, t) \triangleq \mathcal{R}_1(u, t) \cup \cdots \cup \mathcal{R}_M(u, t)$.

Proof: Given in Section 5. ◇

3 Caching policies and the optimality

Consider infinitely many documents of unit size out of which x can be stored in a local memory called cache. When an item is requested, the cache is searched first and we say that there is a cache hit if the item is found in the cache. In this case the cache content is left unchanged. Otherwise, we say that there is a cache fault/miss and the missing item is brought in from the outside world. At the time of a fault, a decision whether to replace some item from the cache with a missing item has to be made. We assume that replacements are optional, i.e., the cache content can be left unchanged even in the case of fault. A caching algorithm represents a set of replacement rules. We consider a class of caching algorithms whose information decisions are made using only the information of past and present requests and past decisions.

More formally, let $\mathcal{C}_t^\pi$ be a cache content at time t under policy π. When the request for a document R_n is made, the cache with content $\mathcal{C}_{\tau_n}^\pi$, $n = 0, 1, \dots$, is searched first. If document R_n is already in the cache ($R_n \in \mathcal{C}_{\tau_n}^\pi$), then we use the convention that no document is replaced. On the other hand, if document R_n is not an element of $\mathcal{C}_{\tau_n}^\pi$, then a document to be replaced is chosen from a set $\mathcal{C}_{\tau_n}^\pi \cup \{R_n\}$ using a particular eviction policy. At any moment of request, τ_n, the decision what to replace in the cache is based on $R_1, R_2, \dots, R_n, \mathcal{C}_{\tau_0}^\pi, \mathcal{C}_{\tau_1}^\pi, \dots, \mathcal{C}_{\tau_n}^\pi$. Note that this information already contains all the replacement decisions made up to time τ_n. This is the same information as the one used in the Markov decision framework [5].

The set of the previously described cache replacement policies, say $\mathcal{P}_c$, is quite large and contains mandatory caching rules (more typical for a computer memory environment). Furthermore, the set $\mathcal{P}_c$ also contains the static algorithm, that places a fixed collection of documents $\mathcal{C}_t^\pi \equiv \mathcal{C}$ in the cache, and, after this selection is made, the content of the cache is never changed.

Now, define the long-run cache fault probability corresponding to the policy $\pi \in \mathcal{P}_c$ and a cache of size x as

$$P(\pi, x) \triangleq \limsup_{T\to\infty} \frac{\mathbb{E}\left[\sum_{\tau_n \in [0,T]} 1[R_n \notin \mathcal{C}_{\tau_n}^\pi]\right]}{T}, \tag{3.3}$$

recall that $\mathbb{E}N(0,T) = T$. Note that we use the $\limsup$ in this definition since the limit may not exist in general.

Next, we show that

$$P(\pi, x) = \limsup_{k\to\infty} \frac{\mathbb{E}\left[\sum_{\tau_n \in (0,\mathcal{T}_k)} 1[R_n \notin \mathcal{C}_{\tau_n}^\pi]\right]}{\mathbb{E}N(0,\mathcal{T}_k)}, \tag{3.4}$$

where $\mathcal{T}_k$ are the regenerative points, as defined in the previous section. To this end, for the lower bound, for any $0 < \epsilon < 1$, let $k \equiv k(T,\epsilon) \triangleq \lfloor T(1-\epsilon)/\mathbb{E}[\mathcal{T}_2 - \mathcal{T}_1]\rfloor$, where $\lfloor u \rfloor$ is the largest integer that is less or equal to u. Then, note that

$$\begin{aligned}
&\frac{1}{T}\mathbb{E}\left[\sum_{\tau_n \in (0,\mathcal{T}_k)} 1[R_n \notin \mathcal{C}_{\tau_n}^\pi]\right] \\
&\geq \mathbb{E}\left[1[\mathcal{T}_k < T]\frac{\sum_{\tau_n \in (0,\mathcal{T}_k)} 1[R_n \notin \mathcal{C}_{\tau_n}^\pi]}{T}\right] \\
&\geq \mathbb{E}\left[\frac{\sum_{\tau_n \in (0,\mathcal{T}_k)} 1[R_n \notin \mathcal{C}_{\tau_n}^\pi]}{T}\right] - \mathbb{E}\left[1[\mathcal{T}_k > T]\frac{N(0,T)}{T}\right].
\end{aligned} \tag{3.5}$$

Next, using the Weak Law of Large Numbers for $\mathbb{P}[\mathcal{T}_k > T]$ (as $T \to \infty$) and the fact that $N(0,T)$ is Poisson with mean T in the preceding inequality, we obtain

$$\begin{aligned}
P(\pi, x) &\geq (1-\epsilon) \limsup_{\substack{T\to\infty \\ k=\lfloor \frac{T(1-\epsilon)}{\mathbb{E}[\mathcal{T}_2-\mathcal{T}_1]}\rfloor}} \frac{\mathbb{E}\left[\sum_{\tau_n \in (0,\mathcal{T}_k)} 1[R_n \notin \mathcal{C}_{\tau_n}^\pi]\right]}{\mathbb{E}N(0,\mathcal{T}_k)} \\
&= (1-\epsilon) \limsup_{k\to\infty} \frac{\mathbb{E}\left[\sum_{\tau_n \in (0,\mathcal{T}_k)} 1[R_n \notin \mathcal{C}_{\tau_n}^\pi]\right]}{\mathbb{E}N(0,\mathcal{T}_k)},
\end{aligned}$$

since the set $\{k : k = \lfloor T(1-\epsilon)/\mathbb{E}[\mathcal{T}_2 - \mathcal{T}_1]\rfloor, T > 0\}$, covers all integers. We complete the proof of the lower bound by passing $\epsilon \to 0$. Upper bound uses similar arguments and we omit the details.

Next, observe the static policy s, where $\mathcal{C}_{\tau_n}^\pi \equiv \{1, 2, \dots, x\}$ for every n. Then, due to ergodicity of the request process

$$P_s(x) \triangleq P(s, x) = \sum_{i > x} q_i.$$

Since the static policy belongs to the set of caching algorithms $\mathcal{P}_c$, we conclude that

$$P_s(x) \geq \inf_{\pi \in \mathcal{P}_c} P(\pi, x). \tag{3.6}$$

Our goal in this paper is to show that for large cache sizes x there is no caching policy that performs better, i.e., achieves long-term fault probability smaller than $P_s(x)$. This is stated in the following main result of this paper.

THEOREM 3.1. *For the request process defined in Section 2, the static policy that stores documents with the largest marginal popularities minimizes the long-term cache fault probability for large cache sizes, i.e.,*

$$\inf_{\pi \in \mathcal{P}_c} P(\pi, x) \sim P_s(x) \ \text{ as } \ x \to \infty. \tag{3.7}$$

Remarks: (i) From the examination of the following proof it is clear that the result holds for any regenerative request process that satisfies Lemma 2.1. (ii) Though asymptotically long-term optimal, static frequency rule possesses other undesirable properties such as high complexity and non-adaptability to variations in the request patterns. However, its optimal performance presents an important benchmark for evaluating and comparing widely implemented caching policies in the Web environment.

Proof: In view of (3.6), we only need to show that $\inf_{\pi\in\mathcal{P}_c} P(\pi,x)$ is asymptotically lower bounded by $P_s(x)$ as $x\to\infty$.

For any set $\mathcal{A}$, let $|\mathcal{A}|$ denote the number of elements in $\mathcal{A}$ and $\mathcal{A}\setminus\mathcal{B}$ represent the set difference. Then, it is easy to see that the number of cache faults in $[t,u)$, $t<u$, is lower bounded by $|\mathcal{R}(t,u)\setminus\mathcal{C}_t^\pi|$ since every item that was not in the cache at time t results in at least one fault when requested for the first time; in particular, if $t=\mathcal{T}_j$, $u=\mathcal{T}_{j+1}$,

$$\sum_{\tau_n\in[\mathcal{T}_j,\mathcal{T}_{j+1})} 1[R_n\notin\mathcal{C}^\pi_{\tau_n}]\ge|\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})\setminus\mathcal{C}^\pi_{\mathcal{T}_j}|. \tag{3.8}$$

This inequality and (3.4) results in

$$P(\pi,x)\ge\limsup_{k\to\infty}\frac{1}{\mathbb{E}N(0,\mathcal{T}_k)}\sum_{j=1}^{k-1}\mathbb{E}[|\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})\setminus\mathcal{C}^\pi_{\mathcal{T}_j}|]. \tag{3.9}$$

Now, since we consider caching policies where replacement decisions depend only on the previous cache contents and requests, due to renewal structure of the request process we conclude that for every $j\ge1$ and all $i\ge1$, events $\{i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})\}$ and $\{i\in\mathcal{C}^\pi_{\mathcal{T}_j}\}$ are independent. Therefore, for every $j\ge1$,

$$\begin{aligned}
&\mathbb{E}\left[|\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})\setminus\mathcal{C}^\pi_{\mathcal{T}_j}||\mathcal{C}^\pi_{\mathcal{T}_j}=\mathcal{C}\right]\\
&=\sum_{i\ge1}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1}),i\notin\mathcal{C}^\pi_{\mathcal{T}_j}|\mathcal{C}^\pi_{\mathcal{T}_j}=\mathcal{C}]\\
&=\sum_{i\ge1}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]1[i\notin\mathcal{C}]\\
&=\sum_{i\notin\mathcal{C}}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})].
\end{aligned}$$

Thus, for any $j\ge1$,

$$\mathbb{E}[|\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})\setminus\mathcal{C}^\pi_{\mathcal{T}_j}|]\ge\inf_{\mathcal{C}:|\mathcal{C}|=x}\sum_{i\notin\mathcal{C}}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]. \tag{3.10}$$

Next, we show that the cache content $[1,x]\triangleq\{1,\dots,x\}$ achieves the infimum in the previous expression for large cache sizes. This is equivalent to proving that, as $x\to\infty$,

$$\inf_{\mathcal{C}:|\mathcal{C}|=x}\sum_{i\notin\mathcal{C}}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]\gtrsim\sum_{i\notin[1,x]}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]. \tag{3.11}$$

We will justify the previous statement by showing that for any set $\mathcal{C}$ obtained from $[1,x]$ by placing documents from the set $\{x+1,\dots\}$ instead of those in $[1,x]$ can not result in $\sum_{i\notin\mathcal{C}}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]<\sum_{i\notin[1,x]}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]$ for large cache sizes x.

Lemma 2.1 implies that for an arbitrarily chosen $\epsilon>0$ there exists finite integer i_0 such that for all $i\ge i_0$

$$\begin{aligned}
(1-\epsilon)q_i\mathbb{E}[\mathcal{T}_{j+1}-\mathcal{T}_j]&<\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]\\
&<(1+\epsilon)q_i\mathbb{E}[\mathcal{T}_{j+1}-\mathcal{T}_j].
\end{aligned} \tag{3.12}$$

Thus, using the previous expression and $q_i\downarrow0$ as $i\to\infty$, we conclude that for all $k\le i_0$ there exists $x_0\ge i_0$ large, such that for all $i\ge x_0$

$$\min_{1\le k\le i_0}\mathbb{P}[k\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]>\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]. \tag{3.13}$$

Now, assume that the cache is of size $x\ge x_0$ and observe different cache contents $\mathcal{C}$ obtained from $[1,x]$ by replacing its documents with items from $\{x+1,x+2,\dots\}$. Next, using (3.13), we conclude that replacing documents enumerated with $\{1,\dots,i_0\}$ can only increase the sum on the left hand side of (3.11). On the other hand, observe cache contents $\mathcal{C}$ that are obtained from $[1,x]$ by replacing documents enumerated as $\{i_0+1,\dots,x\}$ with items from $\{x+1,\dots\}$. Then, it is easy to see that proving inequality (3.11) is equivalent to showing that $\sum_{i\in[i_0+1,x]}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]\ge\sum_{i\in\mathcal{C}\setminus[1,i_0]}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]$. Next, since for any $i\ge i_0$ inequalities (3.12) hold, we conclude

$$\begin{aligned}
\frac{\sum_{i\in[i_0+1,x]}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]}{\sum_{i\in\mathcal{C}\setminus[1,i_0]}\mathbb{P}[i\in\mathcal{R}(\mathcal{T}_j,\mathcal{T}_{j+1})]}&\ge\frac{(1-\epsilon)\sum_{i\in[i_0+1,x]}q_i}{(1+\epsilon)\sum_{i\in\mathcal{C}\setminus[1,i_0]}q_i}\\
&\ge\frac{1-\epsilon}{1+\epsilon},
\end{aligned}$$

where the second inequality in the previous expression follows from the monotonicity of q_is. Then, by passing $\epsilon\to0$ we prove inequality (3.11).

Note that after applying the lower bound (3.11) in (3.10), in conjunction with (3.9), the renewal nature of the regenerative points and Lemma 2.1, we obtain that as $x\to\infty$

$$\inf_{\pi\in\mathcal{P}_c}P(\pi,x)\gtrsim\sum_{i\ge x}q_i, \tag{3.14}$$

which completes the proof of the theorem. ◇

4 Further extensions and concluding remarks

In this paper we prove that the static frequency rule minimizes the long term fault probability, for large cache sizes, in the presence of correlated requests. Although the frequency algorithm and its practical version the LFU policy is not commonly used in practice due to their complexity and static nature, our result provides a benchmark for evaluating the popular heuristic schemes. In order to capture dependency in the request patterns, we use semi-Markov modulation technique, which is capable of modeling a wide range of statistical correlation, including the LRD that was repeatedly experimentally observed in Web access patterns.

There are several generalizations of our results that are worth mentioning. First, the definition of the fault probability in (3.4) can be generalized by replacing terms $1[R_n \notin \mathcal{C}^{\pi}_{\tau_n}]$ with $f(R_n)1[R_n \notin \mathcal{C}^{\pi}_{\tau_n}]$, where $f(i)$ could represent the cost of retrieving document i, e.g., the delay of fetching item i. Then, using basically the same arguments as in the proof of Theorem 3.1, one can easily show that a static policy which maximizes $\sum_{i=1}^{x} f(i)q_i$ is asymptotically optimal.

Second, in the context of documents with different sizes, in view of Section 4.1 of [10] and the arguments from the proof of Theorem 3.1, one can prove the following result:

THEOREM 4.1. *Assume that there are $D < \infty$ different document sizes. Then, if marginal request distribution is long tailed, i.e., $q_i \sim q_{i+k}$ as $i \to \infty$ for any finite integer k, the static rule that places documents with the largest ratio q_i/s_i, subject to the constraint $\sum_i s_i \le x$, is asymptotically optimal.*

Finally, in light of our recent result on the asymptotic performance of the ordinary LRU caching rule in the presence of semi-Markov modulated requests and Zipf's law marginal distributions ($q_i \sim c/i^{\alpha}$ as $i \to \infty$, $c > 0$) obtained in Theorem 3 of [9], asymptotic optimality of the static frequency rule implies that the LRU is factor $e^{\gamma} \approx 1.78$ away from the optimal (γ is the Euler constant, i.e. $\gamma \approx 0.57721\dots$). Therefore, in view of other desirable properties, such as self-organizing nature and low complexity, the LRU rule has excellent performance even in the presence of statistically correlated requests.

5 Proof of Lemma 2.1

In this section, we prove the asymptotic relation (2.2) stated at the end of Section 2.

Note that

$$\begin{aligned}\mathbb{P}[i \in \mathcal{R}(\mathcal{T}_1,\mathcal{T}_2)] &= 1 - \mathbb{P}[i \notin \mathcal{R}_1(\mathcal{T}_1,\mathcal{T}_2),\dots, i \notin \mathcal{R}_M(\mathcal{T}_1,\mathcal{T}_2)]\\ &= \mathbb{E}[1-(1-q_i^{(1)})^{N_1}\dots(1-q_i^{(M)})^{N_M}],\end{aligned} \tag{5.15}$$

where $N_r \triangleq N_r(\mathcal{T}_1,\mathcal{T}_2)$, $1 \le r \le M$. Then, since $q_i \to 0$ as $i \to \infty$, it follows that $q_i^{(r)} \to 0$ as $i \to \infty$, $1 \le r \le M$. In addition, $1 - e^{-x} \le x$ for all $x \ge 0$ and for any $1 > \epsilon > 0$, there exists $x_0(\epsilon) > 0$, such that for all $0 \le x \le x_0(\epsilon)$ inequality $1 - x \ge e^{-x(1+\epsilon)}$ holds, and, therefore, for i large enough

$$\begin{aligned}&\mathbb{E}\left[1 - e^{-(q_i^{(1)}N_1+\dots+q_i^{(M)}N_M)}\right]\\ &\le \mathbb{E}\left[1-(1-q_i^{(1)})^{N_1}\dots(1-q_i^{(M)})^{N_M}\right]\\ &\le \mathbb{E}\left[1 - e^{-(1+\epsilon)(q_i^{(1)}N_1+\dots+q_i^{(M)}N_M)}\right].\end{aligned} \tag{5.16}$$

Then, since $1 - e^{-x} \le x$ for $x \ge 0$, we obtain, for i large enough,

$$\begin{aligned}&\mathbb{E}\left[1 - e^{-(1+\epsilon)(N_1q_i^{(1)}+\dots+N_Mq_i^{(M)})}\right]\\ &\le (1+\epsilon)\mathbb{E}\left[q_i^{(1)}N_1+\dots+q_i^{(M)}N_M\right].\end{aligned} \tag{5.17}$$

Next, let $N \triangleq N_1 + \dots + N_M$. Then, we show that $q_i^{(1)}\mathbb{E}N_1 + \dots + q_i^{(M)}\mathbb{E}N_M = q_i\mathbb{E}N$. From ergodicity of the process J_t, it follows

$$\pi_r = \frac{\mathbb{E}\mathcal{T}_{1r}}{\mathbb{E}[\mathcal{T}_2 - \mathcal{T}_1]},$$

where $\mathcal{T}_{1r}$, $1 \le r \le M$, is the length of time that J_t spends in state r during the renewal interval $(\mathcal{T}_1, \mathcal{T}_2)$ (see Section 1.6 of [4]). Finally, using $\mathbb{E}N = \mathbb{E}[\mathcal{T}_2 - \mathcal{T}_1]$ and $\mathbb{E}N_r = \mathbb{E}\mathcal{T}_{1r}$, $1 \le r \le M$ (Poisson process of rate 1), in conjunction with (5.17), we conclude, for i large

$$\mathbb{E}\left[1 - e^{-(1+\epsilon)(N_1q_i^{(1)}+\dots+N_Mq_i^{(M)})}\right] \le (1+\epsilon)q_i\mathbb{E}[\mathcal{T}_2 - \mathcal{T}_1]. \tag{5.18}$$

Next, we estimate the lower bound in (5.16). After conditioning we obtain

$$\begin{aligned}&\mathbb{E}\left[1 - e^{-(N_1q_i^{(1)}+\dots+N_Mq_i^{(M)})}\right]\\ &\ge \mathbb{E}\left[\left(1 - e^{-(N_1q_i^{(1)}+\dots+N_Mq_i^{(M)})}\right)1\left[N \le \bar{q}_i^{-\frac{1}{2}}\right]\right],\end{aligned} \tag{5.19}$$

where $q_i^{(r)} \le \bar{q}_i \triangleq \frac{q_i}{\min_r \pi_r} \le Hq_i$, $1 \le r \le M$. Then, note that for every $\omega \in \{N \le \bar{q}_i^{-\frac{1}{2}}\}$, $q_i^{(1)}N_1 + \dots + q_i^{(M)}N_M \le \frac{\bar{q}_i}{\sqrt{\bar{q}_i}} = \sqrt{\bar{q}_i}$. In addition, for any $1 > \epsilon > 0$, there exists $x_0(\epsilon) > 0$, such that for all $0 \le x \le x_0(\epsilon)$ inequality $1 - e^{-x} \ge (1-\epsilon)x$ holds, and, therefore, for i large enough such that $\sqrt{\bar{q}_i} \le x_0(\epsilon)$

$$\begin{aligned}&\mathbb{E}\left[\left(1 - e^{-(N_1q_i^{(1)}+\dots+N_Mq_i^{(M)})}\right)1\left[N \le \bar{q}_i^{-\frac{1}{2}}\right]\right]\\ &\ge (1-\epsilon)\mathbb{E}\left[(N_1q_i^{(1)}+\dots+N_Mq_i^{(M)})1\left[N \le \bar{q}_i^{-\frac{1}{2}}\right]\right]\\ &\ge (1-\epsilon)q_i\mathbb{E}[\mathcal{T}_2 - \mathcal{T}_1] - (1-\epsilon)\bar{q}_i\mathbb{E}[N1[N > \bar{q}_i^{-\frac{1}{2}}]].\end{aligned}$$

Then, $\mathbb{E}[N1[N > \bar{q}_i^{-\frac{1}{2}}]] \to 0$ as $i \to \infty$ since $1/\sqrt{\bar{q}_i} \to \infty$ as $i \to \infty$ and $\mathbb{E}N < \infty$ and, therefore, in conjunction with (5.19), we obtain, as $i \to \infty$,

$$\mathbb{E}\left[1 - e^{-(N_1q_i^{(1)}+\dots+N_Mq_i^{(M)})}\right] \gtrsim (1-\epsilon)q_i\mathbb{E}[\mathcal{T}_2 - \mathcal{T}_1].$$

Finally, after letting $\epsilon \to 0$ in the previous expression and using (5.18), we complete the proof of this lemma. $\diamond$

References

[1] M. Abrams and R. Wooster. Proxy caching that estimates edge load delays. In *Proceedings of 6th Int. World Wide Web Conf.*, Santa Clara, CA, April 1997.

[2] V. Almeida, A. Bestavros, M. Crovella, and A. de Oliviera. Characterizing reference locality in the WWW. In *Proceedings of the Fourth International Conference on Parallel and Distributed Information Systems*, Miami Beach, Florida, December 1996.

[3] M. Arlitt and C. Williamson. Web server workload characteristics: The search for invariants. In *Proceedings of ACM SIGMETRICS*, May 1996.

[4] F. Baccelli and P. Brémaud. *Elements of Queueing Theory*. Springer–Verlag, 2002.

[5] O. Bahat and A. M. Makowski. Optimal replacement policies for non-uniform cache objects with optional eviction. In *Proceedings of Infocom 2003*, San Francisco, California, USA, April 2003.

[6] L. Breslau, Pei Cao, Li Fan, G. Phillips, and S. Shenker. Web caching and Zipf-like distributions: Evidence and implications. In *IEEE INFOCOM*, 1999.

[7] P. Cao and S. Irani. Cost-aware WWW proxy caching algorithms. In *Proceedings of the USENIX 1997 Annual Technical Conference*, Anaheim, California, January 1997.

[8] P. R. Jelenković and A. Radovanović. Asymptotic insensitivity of least-recently-used caching to statistical dependency. In *Proceedings of INFOCOM 2003*, San Fransisco, April 2003.

[9] P. R. Jelenković and A. Radovanović. Least-recently-used caching with dependent requests. *Theoretical Computer Science*, 326:293–327, 2004.

[10] P. R. Jelenković and A. Radovanović. Optimizing LRU for variable document sizes. *Combinatorics, Probability & Computing*, 13:1–17, 2004.

[11] Shudong Jin and Azer Bestavros. GreedyDual* Web Caching Algorithm. In *Proceedings of the 5th International Web Caching and Content Delivery Workshop*, Lisbon, Portugal, May 2000.

[12] Shudong Jin and Azer 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 Fransisco, CA, August 2000.

[13] E. G. Coffman Jr. and P. J. Denning. *Operating Systems Theory*. Prentice-Hall, 1973.