A Proof for Effective and Efficient Web Caching

G. Karakostas

Telcordia Technologies Inc.

Morristown, NJ 07960

USA

D.N. Serpanos **(Contact Author)**

Dept. of Eletrical and Computer Eng.

Univ. of Patras

GR-26500 Rion, Patras

Greece

E-mail: serpanos@ee.upatras.gr

A Proof for Effective and Efficient Web Caching

G. Karakostas and D.N. Serpanos

Abstract

Perfect-LFU, the optimal replacement policy for Web caches, achieves high performance (cache hit-rate) exploiting Zipf's law, but is very expensive to implement. We prove that, an alternative policy, called Window-LFU, achieves equivalent performance at a significantly lower cost.

1 Introduction

Web caches take advantage of Zipf's Law, which governs user requests, in general. The optimal replacement policy is Perfect-LFU (P-LFU) [2], which replaces Web objects based on their popularity. P-LFU is expensive to implement, because correct calculation of popularity requires statistics on all objects accessed in the whole past.

One can make replacement decisions based on Web object popularity in a recent past, called *recent time window*. This policy, called *Window-LFU (W-LFU)*, clearly has a significantly lower implementation cost. In this paper, we prove that, W-LFU achieves high hit rates, equivalent to P-LFU, for small window sizes. Our only assumptions are Zipf's law and statistical independence among user requests.

2 Model and Notation

We consider an enterprise network (or LAN) connected to the Internet through a gateway with a cache, as shown in Figure 1. User (client) requests are either served by the cache, if the requested objects are cache-resident, or forwarded to the Internet.

We assume that the set of all available objects is $O = \{O_1, O_2, \ldots, O_N\}$, with size $\mid O \mid = N$, and that client requests obey Zipf's law. So, we model the stream of client requests, R, as a series of *independent* trials drawn from a Zipf distribution over the N objects: a new request in R is for the i-th most popular object with probability $P_N(i) = \frac{\Omega}{i}$, where $\Omega = \frac{1}{H_N} \approx \frac{1}{\ln N}$ (H_N is the N-th harmonic number). The system is considered *closed*, i.e., that N and the nature of objects does not change. This consideration is realistic for time intervals in the order of weeks or months, when we observe no significant changes in the population of requested objects.

Caches store and replace objects considering accesses in a recent past, which we call *time window* W. Window W has length $\mid W \mid$, measured in number of requests, and contains the last $\mid W \mid$ requests $W_1, W_2, \ldots, W_{|W|}$; e.g., in Fig. 1, W contains requests $(W_1, W_2, \ldots, W_{|W|}) = (R_k, \ldots, R_{k-|W|+1})$. Analytical results exist only for $\mid W \mid = \mid R \mid$, the case of P-LFU, where R contains all requests of the complete past [5].

We define $n_W(i)$ as the number of requests, in W, for the i-th most popular object (the i-th most popular object is identified based on the number of requests in R). The expected value of $n_W(i)$, denoted $E(i)$, is easily calculated: $E(i) = E[n_W(i)] = \frac{|W|}{i \ln N}$

3 Estimation of Window Size for W-LFU

We need to estimate $\mid W \mid$, so that the achieved cache hit rate approximates the one achieved with P-LFU. We formalize this, through the following definition:

Definition 1 (Good estimator) *W is a* **good estimator** *of the C most popular objects in R, where C is the cache size, if in W:*

1. *the number of requests of the C most popular objects is greater than $E(C+1)$;*

2. *the number of requests of the remaining $N-C$ objects is smaller than $E(C+1)$.*

The definition ensures a separation between the C most popular objects and the $(N-C)$ less popular ones, based on conditions that are too weak to identify the correct ordering of the objects according to their access frequencies (popularities) in the complete history (a stronger condition and analysis appear in [4]). This weakness is the key for our result. The critical observation is that, for the correct implementation of P-LFU, it is sufficient to have the C most popular objects in the cache, without a need to know the specific order of their access frequencies (popularities) in the window.

When the window is *good* (when both conditions are met), W-LFU provides exactly the same hit rate as P-LFU. Our goal is to choose W, so that it is *good* with very high probability. Then, the hit rate of W-LFU will be very close to the one of P-LFU.

In the analysis, we use the following Chernoff bounds from [1]:

Lemma 1 (Chernoff bounds) *Let $X_1, X_2, \ldots, X_n$ be mutually independent random variables such that $Pr[X_i = 1] = p$ and $Pr[X_i = 0] = 1 - p$, for some $p \in [0, 1]$. Let $X = X_1 + X_2 + \ldots + X_n$ and $E[X] = pn$. Then*

$$Pr[X > (1+\theta)pn] \le e^{-\frac{\theta^2}{3}pn} \tag{1}$$

$$Pr[X - pn < -\alpha] < e^{-\alpha^2/2pn} \tag{2}$$

$$Pr[X - pn > \beta] < e^{-2pn/27} \tag{3}$$

$$Pr[X - pn > \gamma] < e^{\gamma - (\gamma + pn)\ln(1+\gamma/pn)} \tag{4}$$

where $0 < \theta \le 1$, $\alpha > 0$, $\beta > 2pn/3$ and $\gamma > 0$.

These bounds describe quantitatively a simple fact: a series of independent trials is concentrated very heavily around its expected value. Using this, we prove that one does not need many trials (past requests) to estimate the expected value (popularity) well.

Theorem 1 *Let $\epsilon > 0$ be any parameter, and C the cache size. Under the assumptions of statistical independence and Zipf's distribution for the requests*

$$H_{W\text{-}LFU}(C) \ge (1-\epsilon) \cdot H_{P\text{-}LFU}(C)$$

where $H_{P\text{-}LFU}(C)$ is the hit rate of P-LFU, and $H_{W\text{-}LFU}(C)$ is the hit rate of W-LFU with $|W| = \max\{\Theta(C^3 \ln C \ln N \ln \frac{1}{\epsilon}), \Theta(C \ln^2 N \ln \frac{1}{\epsilon})\}$

Proof: Assume that the N objects are ordered according to their popularity in R (O_1 is the most popular, O_2 the second most popular, etc.). We define the following sequence of random variables for each O_i:

$$w_j(i) = \begin{cases} 1, & \text{if } W_j \text{ (the } j\text{-th request in } W\text{) is for } O_i \\ 0, & \text{otherwise} \end{cases}, \text{ for } j = 1, 2, \ldots, |W|$$

Then, by hypothesis, the $w_j(i)$'s are mutually independent, $n_W(i) = \sum_{j=1}^{|W|} w_j(i)$ and $Pr[w_j(i) = 1] \approx \frac{1}{i \ln N}$ from Zipf's distribution. We distinguish the following cases:

Case 1: $1 \leq i \leq (C+1)$

From inequality (2) with $\alpha = E(i) - E(C+1)$ we obtain:

$$\begin{aligned} Pr[n_w(i) < E(C+1)] = Pr[n_w(i) - E(i) < -(E(i) - E(C+1))] &< e^{-\frac{|W|(C+1-i)^2}{2i(C+1)^2 \ln N}} \\ &\leq e^{-\frac{|W|}{2C \ln N}} \end{aligned} \tag{5}$$

Case 2: $C+1 < i < 2(C+1)$

From inequality (1) with $\theta = \frac{i-C-1}{C+1}$ we obtain:

$$\begin{aligned} Pr[n_w(i) > E(C+1)] &< e^{-\frac{(i-C-1)^2}{3(C+1)^2} \frac{|W|}{i \ln N}} \\ &\leq e^{-\frac{|W|}{3C(C+1)^2 \ln N}} \end{aligned} \tag{6}$$

Case 3: $2(C+1) \leq i \leq 3(C+1)$

From inequality (3) with $\beta = E(C+1) - E(i)$ we obtain:

$$\begin{aligned} Pr[n_w(i) > E(C+1)] &< e^{-\frac{2|W|}{27i \ln N}} \\ &\leq e^{-\frac{|W|}{40.5(C+1) \ln N}} \end{aligned} \tag{7}$$

Case 4: $3(C+1) < i \leq N$

From inequality (4) with $\gamma = E(C+1) - E(i)$ we obtain:

$$\begin{aligned} Pr[n_w(i) > E(C+1)] &< e^{E(C+1) - E(i) - E(C+1) \ln \frac{i}{C+1}} \\ &\leq e^{-\frac{2|W|}{5(C+1) \ln N}} \end{aligned} \tag{8}$$

The probability that a window is not a good estimator is evaluated as:

$$\begin{aligned} Pr[W \text{ is } \textit{not} \text{ good}] &= Pr[\text{Case 1 holds} \vee \ldots \vee \text{Case 4 holds, for some } i] \\ &\leq Ce^{-\frac{|W|}{2C \ln N}} + Ce^{-\frac{|W|}{3C(C+1)^2 \ln N}} + \\ &+ (C+2)e^{-\frac{|W|}{40.5(C+1) \ln N}} + (N - 3(C+1))e^{-\frac{2|W|}{5(C+1) \ln N}} \end{aligned} \tag{9}$$

The probability in Equation (9) can be forced to be smaller than any constant $\epsilon > 0$, if we choose $|W| = \max\{\Theta(C^3 \ln C \ln N \ln \frac{1}{\epsilon}), \Theta(C \ln^2 N \ln \frac{1}{\epsilon})\}$.

Given the defintions of $H_{\text{W-LFU}}(C, W)$ and $H_{\text{P-LFU}}(C)$ the following relations hold:

$$H_{\text{P-LFU}}(C) = \sum_{i=1}^{C} \frac{1}{i \ln N} \tag{10}$$

$$
\begin{aligned}
H_{\text{W-LFU}}(C,W) &= Pr[\text{next requested item } r \text{ is in cache}] \\
&\geq \sum_{i=1}^{C} Pr[r = i | W \text{ is good estimator}] \times Pr[W \text{ is good estimator}] \\
&= \sum_{i=1}^{C} \frac{1}{i \ln N} \times (1 - Pr[W \text{ is not a good estimator}]) \qquad (11) \\
&\overset{(9)}{\geq} (1-\epsilon) \sum_{i=1}^{C} \frac{1}{i \ln N} \\
&= (1-\epsilon) H_{\text{P-LFU}}(C)
\end{aligned}
$$

where $\epsilon > 0$ is the accuracy constant we have chosen. □

The case of Zipf-like distributions: The proof is essentially the same when the distribution is Zipf-like (e.g., the distributions described in [2]). In this case, the bound of Theorem 1 is $|W| = \max\{\Theta(N^{1-\alpha} C^3 \ln C \ln \frac{1}{\epsilon}), \Theta(N^{2-2\alpha} C \ln \frac{1}{\epsilon})\}$. In practice α is a number between 0.6 and 0.9 (cf. [2]), so the window size depends *sublinearly* on N.

Generalization: The previous analysis does not use the fact that the window contains the most recent requests. Hence, under the same assumptions, it can be generalized to *any* subset of the request history that has size $|W|$. So, our result is a generalization of the LFU policies that are based on the most recent past and are already in use. In practice, the use the most recent requests leads to easy implementation.

4 Discussion of the Results

W-LFU is an effective replacement policy, because it achieves performance equivalent to P-LFU. It is also efficient for two reasons, as analysis showed. First, its required window size depends only on the logarithm of N (the effect of N on the replacement policy is reduced exponentially[1]). This is significant, because N is not a controllable parameter.

Second, W-LFU has a low implementation cost, because it keeps statistics on a number of objects which is not larger than $|W|$. Although Case 2 shows a dependency on the cache size C, which may be impractical for big enough caches, the result is very strong (approximation of P-LFU within any constant factor). This means that, in practice, smaller window sizes should perform quite well, e.g., $|W| = O(C \ln N)$ or

[1] or to sublinearity, in case of Zipf-like distributions

$O(C^2 \ln N)$. This has been verified through simulation with real traces [3].

References

[1] N. Alon, J. Spencer, and P. Erdos. *The Probabilistic Method.* J.Wiley & Sons, 1992.

[2] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web Caching and Zipf-like Distributions: Evidence and Implications. In *Proceedings of Infocom'99*, 1999.

[3] G. Karakostas and D.N. Serpanos. Practical LFU Implementation for Web Caching. Technical Report TR-622-00, Dept. of Computer Science, Princeton University, June 2000.

[4] D.N. Serpanos, G. Karakostas, and W.H. Wolf. Effective Caching of Web Objects Using Zipf's Law. In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2000)*, July, 30 - August, 2 2000.

[5] D.N. Serpanos and W.H. Wolf. Caching Web Objects Using Zipf's Law. In *Proceedings of SPIE, Vol. 3527, Photonics East, Technical Conference 3527: Multimedia Storage and Archiving Systems III, Boston, MA, USA, November 2-4*, 1998.

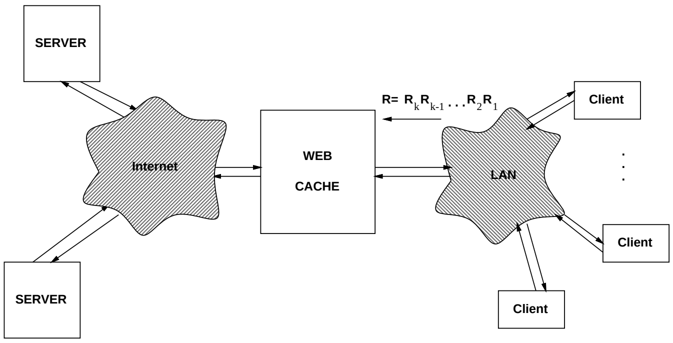

Figure 1: A simple caching environment