Approximate Analysis of LRU in the Case of Short Term Correlations

Antonis Panagakis, Athanasios Vaios, and Ioannis Stavrakakis

Dept. of Informatics & Telecommunications
National & Kapodistrian University of Athens
Panepistimiopolis, 157 84 Ilissia, Athens, Greece

Abstract

One of the most widely considered cache replacement policies is Least Recently Used (LRU) based on which many other policies have been developed. LRU has been studied analytically in the literature under the assumption that the object requests are independent. However, such an assumption does not seem to be in agreement with recent studies of Web-traces, which indicate the existence of short term correlations among the requests. This paper introduces an approximate analysis that fairly accurately predicts the hit ratio of the LRU policy in the case of short term correlations. The approximation approach is based on the relation between the working set model and LRU, while the request generation process is assumed to follow a recently proposed model for Web-traces, which captures short term correlations among the requests. The accuracy of the introduced approximate analysis is validated for synthetic as well as real Web-traces.

Key words: Caching, LRU, correlated requests

Email address: {apan, avaios, ioannis}@di.uoa.gr (Antonis Panagakis,

1 Introduction

Web caching is a widely deployed technique aiming at reducing network bandwidth usage, content access delay and origin server loads. Among the issues that concern the design of an efficient Web caching system like the exact location of the proxies, the cooperation among the proxies or the cache resolution/routing process ([14]), the cache replacement policy to be applied (i.e. the process responsible for making the decisions on the eviction of objects in case the cache is full) has attracted a great deal of research interest. Several replacement policies have been proposed varying in both the achieved efficiency and the induced complexity ([2], [11]). One of the most widely considered cache replacement policies is Least Recently Used (LRU), on the basis of which many other policies have been developed; according to LRU, the object that is evicted at the time a replacement is required is the one that was requested the least recently.

In most of the works regarding the performance analysis of web caching (e.g., [5], [9]), the assumption that the object requests are independent has been adopted. However, the independent reference model has been criticized for inadequately explaining the locality of the requests (the fact that a requested object is highly probable to be requested again in the near future). More specifically, although the assumption of a Zipf-like distribution of the popularity of the objects implies some degree of locality ([3]), recent studies reveal the existence of correlations among the requests in real Web traces ([12], [8]).

This paper presents an approximate analysis for predicting the hit ratio (per-

Athanasios Vaios, and Ioannis Stavrakakis).

centage of the objects being found in the cache when requested) of LRU under the assumption that the request generation process follows a recently proposed model for Web-traces ([12]), which captures short term correlations among the requests. The model has a memory of length h; it is assumed that the request generation process is a probabilistic mixture of a memoryless process and a process that chooses one of the h most recent requests.

An additional interesting finding of this study is that when the memory of the request generation process is less than the capacity of the cache ($h \leq C$), the hit ratio of the LRU replacement policy is approximately independent of the exact length of the memory h. In addition, in this case ($h \leq C$) the value of the hit ratio can be analytically evaluated by utilizing the proposed approximate analysis for $h = 1$.

Finally, it is worth noticing that the LRU policy (that has been mostly concerned for web-caching applications and peer-to-peer networks ([13])) may be applied in other networking paradigms, like that of Delay Tolerant Networks (DTNs), where the mobile nodes exchange their messages upon their encounters and a replacement policy – like LRU – is applied for buffer management ([10]); recently, it was observed that the intermeeting times between the nodes of a DTN are correlated, ([4]). Thus, this work may be used as the basis for further investigation in order to capture the effects of caching on such a new environment.

2 Model description

The popularity of the (equally sized) objects is assumed to follow a Zipf-like distribution. Assuming, without loss of generality, that the N objects are enumerated with decreased popularity order, the probability of object k to be requested is given by

$$q_k = \frac{K}{k^a}, 1 \leq k \leq N, \tag{1}$$

where $K = \left(\sum_{k=1}^{N} \frac{1}{k^a}\right)^{-1}$ is the normalization constant and a is the skewness parameter of the Zipf-like distribution. [1]

In order to capture short term correlations among the requests, we use the model recently proposed in [12], which was shown to fairly accurately approximate the behaviour of real Web-traces.

More specifically, let R_n denote the object requested at time $n \geq 1$, and Y_n, $n \geq 1$, be a sequence of random variables that are independent and identically

[1] As reported in [6], for Web proxies, the value of a is typically less than 1, ranging from 0.64 to 0.83, while for Web servers the reported typical value of a is varying between 1.4 and 1.6.

distributed according to $q_i, 1 \leq i \leq N$. For object requests and $n > h$:

$$R_n = \begin{cases} R_{n-1} \text{ with probability } w_1, \\ R_{n-2} \text{ w.p. } w_2, \\ \vdots \\ R_{n-h} \text{ w.p. } w_h, \\ Y_n \quad \text{ w.p. } \beta, \end{cases} \tag{2}$$

$$\text{where } \beta + \sum_{i=1}^{h} w_i = 1. \tag{3}$$

The w_i's are decreasing with respect to i and follow a Zipf-like distribution; let a_h denote the skewness parameter of this distribution. The history of the h most recent requests is kept in order to capture short term (temporal) correlations, while Y_n injects objects that may or may not have been recently requested and captures long-term (object) popularity. As proved in [12], in stationarity, R_n is distributed like Y_n.

3 Analysis

The applied approximation for the computation of the hit ratio under the described in the previous section request arrival model capitalizes on the results of [7] regarding the working set model and its relation to LRU. In [7], a discrete time model is considered where time instants are defined by (the event of) objects' requests (or references [2]) and all objects are assumed to be of the same size. The working set $W(t, T)$ at time t is defined as the set of distinct

[2] The terms request and reference are used interchangeably in this paper.

objects referenced in the time interval $[t-T+1,t]$ and the working set size $w(t,T)$ is defined as the number of objects in $W(t,T)$; the parameter T is referred to as the "window size". The following expressions have been derived for the average working set size $s(T)$ and the fault probability $m(T)$ (probability that the requested object is not contained in the working set):

$$s(T)=\sum_{z=0}^{T-1}(1-F(z)), \tag{4}$$

$$m(T)=1-F(T), \tag{5}$$

where $F(z)=\sum_{i=1}^{N}q_iF_i(z)$, q_i is the relative frequency of references to object i, N is the total number of objects, and $F_i(z)$ is the inter-reference distribution for object i, which is defined as the fraction of inter-reference intervals for object i that are less than or equal to z; the inter-reference interval for object i is defined as the interval between two successive references to object i (e.g., is equal to z if two successive requests for object i take place at t and at $t+z$). Equations (4) and (5) have been derived in [7] under relatively general assumptions (the request process is stationary and the requests are asymptotically uncorrelated).

In order for the working set model to simulate the LRU policy, T should be let to vary so that $W(t,T)$ always contains precisely C objects (where C refers to the capacity of the cache), in which case $W(t,T)$ will include precisely the contents of the LRU cache.

In this paper, for the approximate analysis of LRU, T is approximated by a constant (and is allowed to take non integer values), i.e. T is assumed to remain (approximately) constant when the working set model is used to simulate LRU. Under this assumption, the mean working set size is (approximately) equal to the capacity of the cache. Thus, based on equations (4) and (5), it follows that [3]

$$C \approx s(T) = \sum_{z=0}^{T-1} (1 - F(z)), \tag{6}$$

$$HR \approx F(T), \tag{7}$$

where C and HR denote the capacity and the hit ratio of the cache, respectively. $F(x)$ is given by

$$F(x) = \sum_{j=1}^{N} q_j F_j(x), \tag{8}$$

where $F_j(x)$ is the probability that the inter-reference interval for object j is less than or equal to x requests. $F_j(x)$ can be expressed as

$$F_j(x) = \sum_{k=1}^{x} G_j(k), \tag{9}$$

where

$$G_j(k) = P(\{R_{n+k} = j, \{R_z \neq j, n < z < n + k\}\} | R_n = j) \tag{10}$$

denotes the probability that the inter-reference interval for object j is exactly equal to k requests. For the computation of $G_j(k)$, the recursive expression

$$G_j(k) = \left(1 - \sum_{m=1}^{k-1} G_j(m)\right) H_j(k), \tag{11}$$

which is directly implied by its definition can be employed, where $H_j(k)$ denotes the probability that the inter-reference interval for object j is exactly

[3] It should be noted that it can be easily concluded that the assumptions under which equations (4) and (5) have been derived also hold for the employed request arrival model.

equal to k requests provided that it is greater than $k-1$ requests, or[4]

$$H_j(k) = P(R_{n+k} = j | \{R_n = j, \{R_z \neq j, n < z < n+k\}\}). \tag{12}$$

Thus, in order to determine the hit ratio of the cache, it is sufficient to calculate $H_j(k)$. To this end, the following Lemma will be used (its proof is quoted in the Appendix).

Lemma 1. *Let*

$$c_j(i) \triangleq P(R_{n+i} = j | R_n = j), 1 \leq i \leq h. \tag{13}$$

Then, $c_j(i)$, $1 \leq i \leq h$, can be obtained by solving the set of equations

$$c_j(i) = \beta q_j + w_i + \sum_{k=1, k\neq i}^{h} w_k c_j(|i-k|), 1 \leq i \leq h. \tag{14}$$

Now, let

$$c_j(k,m) \triangleq P(R_{n-m} = j | \{R_n = j, \{R_z \neq j, n+1 \leq z \leq n+k\}\}),$$

$0 \leq k \leq h-2, 1 \leq m \leq h-k-1$. By definition, it holds that $c_j(0,m) = c_j(m), 1 \leq m \leq h-1$ (see equation (13)). Then, $H_j(k)$ may be expressed as

$$H_j(k) = \begin{cases} \beta q_j + w_k + \sum_{t=k+1}^{h} w_t c_j(k-1, t-k), \; 1 \leq k \leq h, \\ \\ \beta q_j, \qquad\qquad\qquad\qquad\qquad\qquad k > h. \end{cases} \tag{15}$$

The computation of the probabilities $H_j(k)$ for $k \leq h$ is rather difficult since it requires the analysis of an h-th order Markov chain. To overcome the above

[4] Throughout this section, when referring to the n-th request it is silently assumed that n is sufficiently large (and definitely $n > h$), since only the steady state behavior of the system is considered.

inefficiency, approximations for $H_j(k)$ are derived in the sequel and lead to an upper bound as well as approximate expressions for the hit ratio.

3.1 Upper bound on the hit ratio

An upper bound on the hit ratio may be obtained by using the quantities $c_j(m)$ instead of $c_j(k,m)$ (it is noted that $c_j(m) \geq c_j(k,m)$, by definition) in equations (7) and (8). More specifically, the following Lemma indicates that by using $\widehat{F}_j(x)$ instead of $F_j(x)$ an upper bound on the HR, denoted as $HR^{\text{upper bound}}$, is obtained (the proof is quoted in the Appendix).

Lemma 2. *Let*

$$\widehat{F}_j(x) \triangleq \sum_{k=1}^{x} \widehat{G}_j(k), x \geq 1,$$

where

$$\widehat{G}_j(k) \triangleq \left(1 - \sum_{m=1}^{k-1} \widehat{G}_j(m)\right) \widehat{H}_j(k), k \geq 1,$$

and

$$\widehat{H}_j(k) \triangleq \begin{cases} \beta q_j + w_k + \sum_{m=k+1}^{h} w_m c_j(m-k), \; 1 \leq k \leq h, \\ \\ H_j(k), \qquad\qquad\qquad\qquad\qquad k > h, \end{cases} \tag{16}$$

where $H_j(k)$ is given by equation (15). Then $F_j(x) \leq \widehat{F}_j(x), \forall x \geq 1$, where $F_j(x)$ is defined by equation (9).

Now, let $\widehat{F}(x) = \sum_{j=1}^{N} q_j \widehat{F}_j(x)$, $\widehat{F}(0) = F(0)$ and let $\widehat{T}$ be the solution of

$$C \approx \sum_{z=0}^{\widehat{T}-1} (1 - \widehat{F}(z)). \tag{17}$$

Then,

$$HR^{\text{upper bound}} = \widehat{F}(\widehat{T}). \tag{18}$$

From Lemma 2, it follows that $\widehat{F}(x) \geq F(x), x \geq 1$ and, by comparing (17) with (6), it is concluded that $\widehat{T} \geq T$ and, thus (see (18), (7)), $HR^{\text{upper bound}} = \widehat{F}(\widehat{T}) \geq HR$.

3.2 Approximate expressions for the hit ratio

In order to provide approximate expressions for the hit ratio, we focus on approximating the probabilities $c_j(k,m)$. To this end, we use the Bayes' rule, according to which:

$$\begin{gathered} P(R_{n-m} = j, R_{n+k+1} \neq j | \{R_n = j, \{R_z \neq j, n+1 \leq z \leq n+k\}\}) = \\ P(R_{n-m} = j | \{R_n = j, \{R_z \neq j, n+1 \leq z \leq n+k+1\}\}) \\ \cdot P(R_{n+k+1} \neq j | \{R_n = j, \{R_z \neq j, n+1 \leq z \leq n+k\}\}) = \\ P(R_{n+k+1} \neq j | \{R_{n-m} = j, R_n = j, \{R_z \neq j, n+1 \leq z \leq n+k\}\}) \\ \cdot P(R_{n-m} = j | \{R_n = j, \{R_z \neq j, n+1 \leq z \leq n+k\}\}). \end{gathered}$$

By employing the above equations on the definition of $c_j(k,m)$ it is obtained that

$$\frac{c_j(k+1,m)}{c_j(k,m)} = \frac{P(R_{n+k+1} \neq j | \{R_{n-m} = j, R_n = j, \{R_z \neq j, n+1 \leq z \leq n+k\}\})}{P(R_{n+k+1} \neq j | \{R_n = j, \{R_z \neq j, n+1 \leq z \leq n+k\}\})}.$$

To simplify the above expression, the approximation

$$\frac{c_j(k+1,m)}{c_j(k,m)} \approx \frac{c_j(1,m)}{c_j(0,m)} = \frac{P(R_{n+1} \neq j | R_{n-m} = j, R_n = j)}{P(R_{n+1} \neq j | R_n = j)} \tag{19}$$

is introduced, which allows to express $c_j(k,m)$ for $k \geq 2$ with respect to $c_j(0,m)$ and $c_j(1,m)$ [5].

[5] As it may be seen from the definition of $c_j(k,m)$, the difference between $c_j(k,m)$ and $c_j(k+1,m)$ lies on the extra knowledge that $R_{k+1} \neq j$; this difference is intu-

The denominator of equation (19) may be expressed as

$$P(R_{n+1} \neq j | R_n = j) = 1 - P(R_{n+1} = j | R_n = j) = 1 - c_1(j).$$

Regarding the numerator of equation (19), it holds that

$$P(R_{n+1} \neq j | \{R_{n-m} = j, R_n = j\}) = 1 - P(R_{n+1} = j | \{R_{n-m} = j, R_n = j\}) =$$
$$1 - (\beta q_j + w_1 + w_{m+1} + \sum_{t=1, t \neq m}^{h-1} w_{t+1} P(R_{n-t} = j | \{R_{n-m} = j, R_n = j\}). \quad (20)$$

By using the inequalities

$$max\{P(R_{n-t} = j | R_n = j), P(R_{n-t} = j | R_{n-m} = j)\} \leq P(R_{n-t} = j | \{R_n = j, R_{n-m} = j\})$$
$$\leq min\{1, P(R_{n-t} = j | R_n = j) + P(R_{n-t} = j | R_{n-m} = j)\} \quad (21)$$

in equation (20), an overestimation and an underestimation are derived for $P(R_{n+1} = j | \{R_{n-m} = j, R_n = j\})$, while a third approximation is obtained as the average of the above two. More specifically, the three approximations, denoted as *ovrest*, *undest* and *est*, are given by the following equations:

(1) Overestimation:

$$P(R_{n+1} = j | \{R_{n-m} = j, R_n = j\})^{ovrest} = \beta q_j + w_1 + w_{m+1}$$
$$+ \sum_{t=1}^{m-1} w_{t+1} \min(1, c_j(m-t) + c_j(t)) + \sum_{t=m+1}^{h-1} w_{t+1} \min(1, c_j(t-m) + c_j(t));$$
$$(22)$$

itively expected to have a rather insignificant impact on the considered probability especially in the case of large values of k (taking also into account that the w_k's are decreasing with k). This intuition was confirmed by the numerical results that are provided in the sequel, at least for the values of $\beta \geq .5$. (This selection of β values is in agreement with the results of [12], where real Web-traces are examined.)

(2) Underestimation:

$$P(R_{n+1}=j|\{R_{n-m}=j,R_n=j\})^{undest}=\beta q_j+w_1+w_{m+1}$$
$$+\sum_{t=1}^{m-1} w_{t+1}\max\left(c_j(m-t),c_j(t)\right)+\sum_{t=m+1}^{h-1} w_{t+1}\max\left(c_j(t-m),c_j(t)\right); \tag{23}$$

(3) Average:

$$P(R_{n+1}=j|\{R_{n-m}=j,R_n=j\})^{est}=$$
$$\frac{P(R_{n+1}=j|\{R_{n-m}=j,R_n=j\})^{undest}+P(R_{n+1}=j|\{R_{n-m}=j,R_n=j\})^{ovrest}}{2}. \tag{24}$$

Starting with equations (22), (23) and (24), three different estimations for the hit ratio are obtained, which are denoted as HR^{ovrest}, HR^{undest} and HR^{est}, respectively. More specifically, the first step is to calculate the probabilities $c_j(0,m)=c_j(m)$ that are obtained by solving the set of equations (14). Equations (19)–(20) in combination with one of (22), (23), or (24) are used in order to obtain the terms $c_j(k,m)$ that are needed to compute $H_j(k)$ from equation (15). Given $H_j(k)$, and by using equations (8)–(11), T is obtained by numerically solving equation (6), while the estimated value of the hit ratio is eventually computed using equation (7).

4 Simulation results

A population of $N=1000$ objects, whose popularity follows a Zipf-like distribution with skewness parameter $a=.8$, is considered. The request arrival process follows the model described in Section 2. Two different values of the history length h ($h = 10$ and $h = 50$) are used, while the w_i's follow a Zipf-like distribution with skewness parameter $a_h = .2$.

Figures 1 and 2 illustrate the simulation results as well as the derived upper bound and estimations for the hit ratio (HR) as a function of the capacity of the cache, for $h = 10$ and $h = 50$, respectively. In each figure, the results for four different values of β are depicted. In Figure 3, the capacity of the cache is set equal to $C = 50$, while the history length h is let to vary. The estimations for the hit ratio are in good agreement with the results obtained through simulations. For relatively larger values of β relatively smaller deviations are observed.

By comparing Figures 1 and 2, it can be observed that the obtained results for $C \geq 50$, that is for values of the cache size that are greater than the maximum value of h that is used, are (almost) identical. This is an indication that the performance of LRU is insensitive to the exact length of the history h as long as $h \leq C$ and depends mainly on β. This is also observed in Figure 3 where it can be seen that for each value of β the hit ratio remains approximately constant for $h \leq C$.

These results are intuitively expected. This is because if the requested object "is in the memory of the process" (one of the h most recently requested objects) it is also located in the cache, as long as the memory of the process is less than the capacity of the cache ($h \leq C$), independently of the exact length of the memory of the process. This means that two different processes with the same value of β (same probability that the requested object "is in the memory of the process") but with different history lengths (that are, however, both less than the capacity of the cache) are expected to approximately lead to the

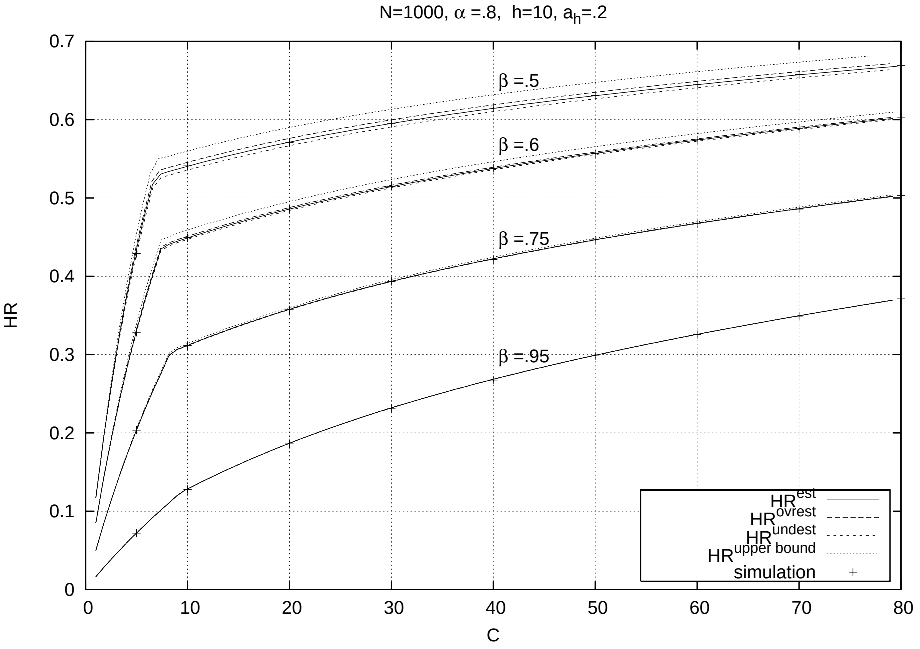

Fig. 1. Estimations, upper bound and simulation results for the hit ratio as a function of the cache size for four different values of β and $h = 10$.

same hit ratio for LRU.

Based on the above observations, one could apply the analysis only for a single value of h lower than the capacity of the cache in order to approximately predict the hit ratio for any value of $h \leq C$. (For simplicity, the analysis for $h = 1$ is preferable and is applied in the sequel.)

4.1 Case: $\mathbf{h} \leq \mathbf{C}$

In order to investigate the idea that the analysis for $h = 1$, which is quoted in the Appendix (see equations (29) and (30)), can be used for the prediction of the hit ratio for any value of $h \leq C$, the analytical results for the hit

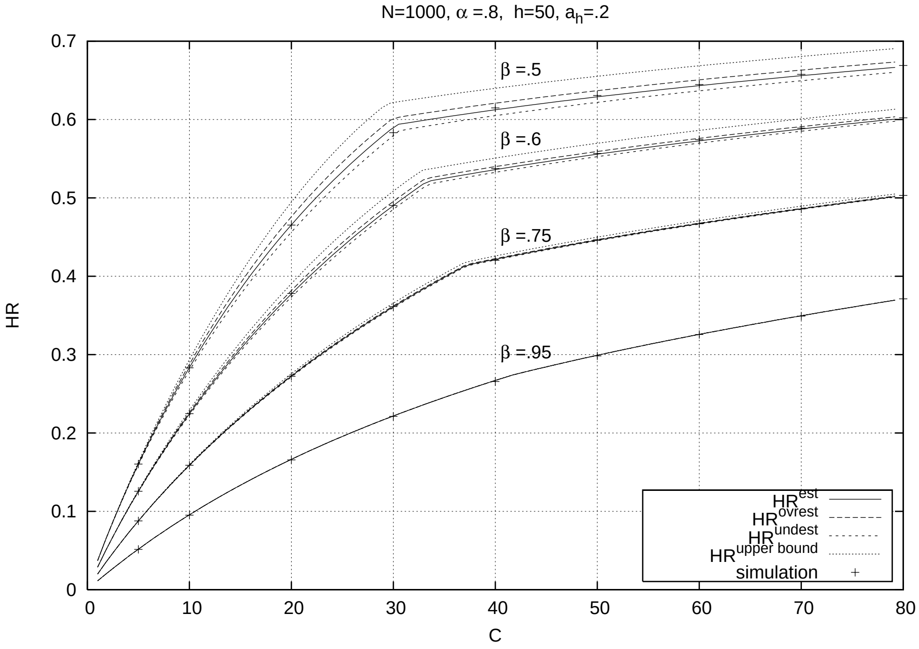

Fig. 2. Estimations, upper bound and simulation results for the hit ratio as a function of the cache size for four different values of β and $h = 50$.

ratio are compared with simulation results for several values of h (and several combinations of the involved parameters).

Table 1 summarizes the parameters used in the simulations. A population of $N = 1000$ objects is considered. For the relative cache size (RCS), which is equal to C/N, since all the objects are assumed to be of unit size, five different values (1%, 2%, 5%, 10%, and 20%) are used and five different values for the skewness parameter of the popularity distribution (0,.2,.4,.8,1.2) are considered[6]. For the request arrival model, five different values are used for

[6] The value of $a = 0$ was used to illustrate the case when the objects have an equal probability to be requested (the popularities are uniformly distributed). As it was shown, the results are similar with the case illustrated in this work ($a = .8$)

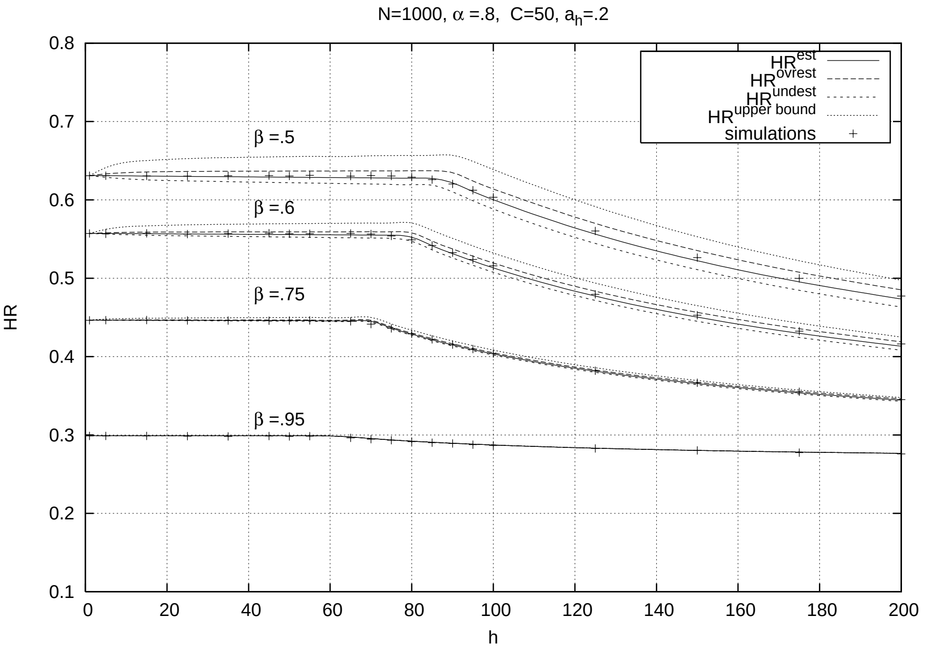

Fig. 3. Estimations, upper bound and simulation results for the hit ratio as a function of the history length h for four different values of β and $C = 50$.

β (.05,.25,.5,.75,.95) and eight different values are used for h (corresponding to a fraction of the cache size equal to .1,.5,.9,1,1.1,1.3,1.5,2). The w_i's ($1 \leq i \leq h$) are assumed to follow a Zipf-like distribution ($w_i = \frac{K_w}{i^{a_h}}$, $1 \leq i \leq h$, $K_w = \left(\sum_{k=1}^{h} \frac{1}{i^{a_h}}\right)^{-1}$) and five different values are used for a_h (0,.2,.4,.8,1.2) (for $a_h = 0$, the w_i's are equal). All the combinations of the aforementioned parameters have been considered.

For all the values of $h/C \leq 1$ and all the combinations of the other parameters, the maximum relative error between the simulation and the analytical results for $h = 1$ is less than 1.7%. Figure 4 illustrates the results for $a = .8$ and

indicating that the conclusions drawn are not limited to only covering the Zipf-like popularity model.

Table 1

Summary of the parameters used in the simulations.

Parameter	Value(s)
N	1000
Relative Cache Size (RCS)	.01,.02,.05,.1,.2
a	0,.2,.4,.8,1.2
h/C	.1,.5,.9,1,1.1,1.3,1.5,2
β	.05,.25,.5,.75,.95
a_h	0,.2,.4,.8,1.2

$RCS = .02$. (The results for the other combinations of these parameters are similar.) In several cases the HR remains approximately constant even for values of $h/C > 1$.

Table 2 summarizes the comparison between simulation results reported in [12] with the analytical results obtained from equations (30) and (29). (In [12], $N = 10000$, $C = 1000$, $h = 100$ and $a_h = .5$ are used for the simulations and, thus, $h \leq C$ holds.) As it may be seen, the relative error between the values derived from the analysis and those from simulations is rather negligible.

Moreover, we experimented with real Web-traces from the National Laboratory for Applied Network Research (NLANR) ([1]). Two different traces are used here from the RTP site (referred to as Trace 1 and 2), which are the longest [7] directly available in NLNAR and concern the 9^{th} and 10^{th} of Jan-

[7] We selected the longest traces in order to validate the analysis for a large number of unique objects (of the order of 10^5), since our simulations refer to an order of 10^3

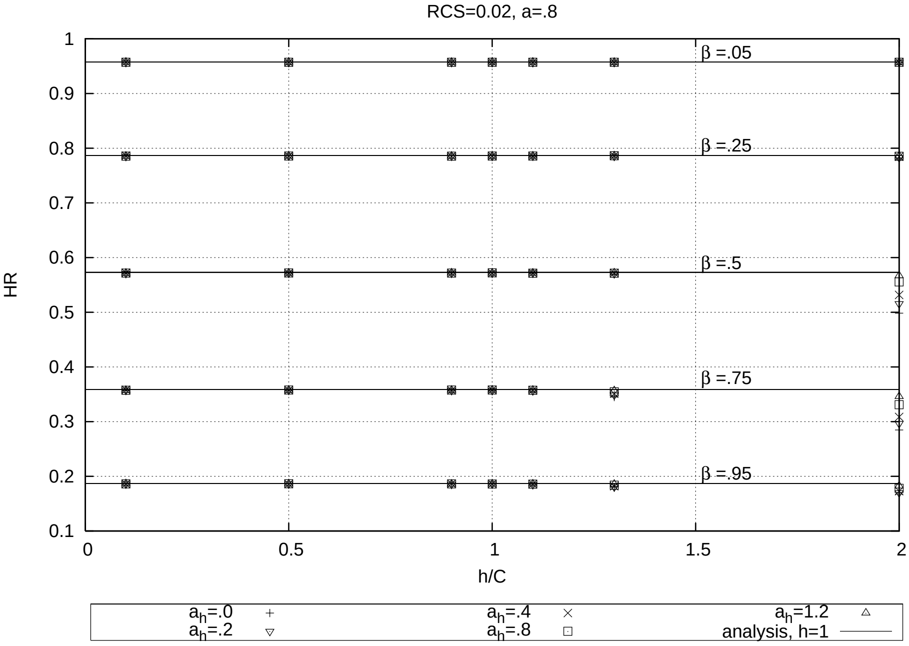

Fig. 4. Comparison between simulation results and analytical results for $h = 1$.

Table 2

Comparison between simulation results reported in [12] and analytical results for $h = 1$.

	$\beta = .5$	$\beta = .75$	$\beta = .95$
[12], simulations	59.01%	38.55%	22.20%
analysis,$h = 1$	59.04%	38.56%	22.17%
Relative error	.051%	.026%	.135%

uary 2007.

Table 3 includes the parameters β and h of the request arrival model that are inferred by the traces along with the hit ratio derived through simulations and

and the ones reported in [12] of 10^4.

Table 3

Characteristics of the real Web-traces used.

	Trace 1	Trace 2
Number of requests	858337	782412
Number of unique objects	152090	164120
β	0.766	0.7403
h	2500	2000
HR for $C = 2h$ (Simulations)	45.48	42.01
HR for $C = 2h$ (Analysis for $h = 1$)	45.45	43.9
HR for $C = 5h$ (Simulations)	58.54	53.48
HR for $C = 5h$ (Analysis for $h = 1$)	57.96	54.45

the analysis for $h = 1$ (for a cache capacity equal to two and five times the history length of the request arrival process, respectively). Figure 5 illustrates the corresponding results concerning Trace 1 for the range from $C = h$ to $C = 6h$.(For Trace 2, the results are similar.)

As it may be seen, the analysis matches the results derived from the real Web-traces. The relative errors comparing with that for synthetic traces are larger for $h \approx C$ but they become negligible when the capacity of the cache is clearly larger than the history length.

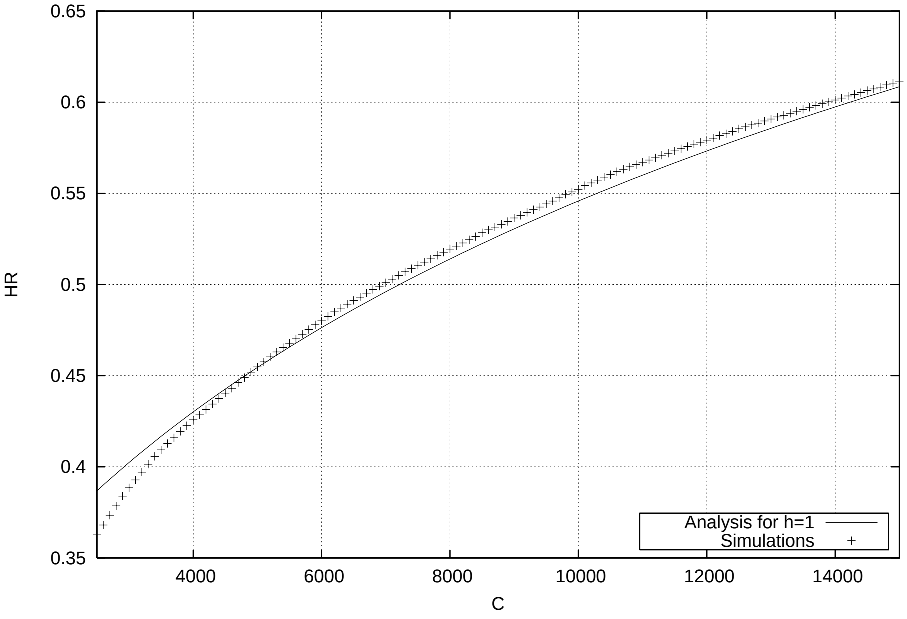

Fig. 5. Comparison between simulations and analytical results (for $h = 1$) for Trace 1.

5 Conclusions

In this paper, an approximate analysis of LRU is derived under the assumption that the request generation process follows a model that captures short term correlations among the requests. More specifically, an upper bound as well as estimations for the hit ratio have been derived. The results indicate that LRU is insensitive to the exact length of the memory of the request generation process as long as the length of the memory is less than the capacity of the cache (which is a rather realistic condition). The analytically derived results are in very good agreement with those obtained for synthetic as well as for real Web-traces.

References

[1] *ftp://ircache.nlanr.net/Traces/.*

[2] Abdullah Balamash and Marwan Krunz. An overview of web caching replacement algorithms. *IEEE Communications Surveys and Tutorials*, 6(2), 2004.

[3] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web caching and zipf-like distributions: Evidence and implications. In *Infocom'99*, 1999.

[4] Augustin Chaintreau, Pan Hui, Jon Crowcroft, Christophe Diot, Richard Gass, and James Scott. Impact of human mobility on opportunistic forwarding algorithms. *IEEE Transactions on Mobile Computing*, 6(6), June 2007.

[5] Hao Che, Zhijung Wang, and Ye Tung. Analysis and design of hierarchical web caching systems. In *Proceedings of the Twentieth Annual Joint Conferenceof the IEEE Computer and Communications Societies (INFOCOM-01)*, pages 1416–1424, Los Alamitos, CA, apr 2001. IEEE Computer Society.

[6] Ludmila Cherkasova and Minaxi Gupta. Characterizing locality, evolution, and life span of accesses in enterprise media server workloads. In *NOSSDAV '02: Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video*, pages 33–42, New York, NY, USA, 2002. ACM Press.

[7] Peter J. Denning and Stuart C. Schwartz. Properties of the working-set model. *Commun. ACM*, 15(3):191–198, 1972.

[8] Shudong Jin and Azer Bestavros. Sources and characteristics of web temporal locality. In *MASCOTS '00: Proceedings of the 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, Washington, DC, USA, 2000. IEEE Computer Society.

[9] Nikolaos Laoutaris, Hao Che, and Ioannis Stavrakakis. The lcd interconnection of lru caches and its analysis. *Performance Evaluation*, 63(7):609–634, 2006.

[10] Christoph Lindemann and Oliver P. Waldhorst. Modeling epidemic information dissemination on mobile devices with finite buffers. In *SIGMETRICS*, pages 121–132, 2005.

[11] Stefan Podlipnig and Laszlo Boszormenyi. A survey of web cache replacement strategies. *ACM Comput. Surv.*, 35(4):374–398, 2003.

[12] Konstantinos Psounis, An Zhu, Balaji Prabhakar, and Rajeev Motwani. Modeling correlations in web traces and implications for designing replacement policies. *Comput. Networks*, 45(4):379–398, 2004.

[13] Osama Saleh and Mohamed Hefeeda. Modeling and caching of peer-to-peer traffic. In *Proc. 14th IEEE International Conference on Network Protocols (ICNP'06)*, 2006.

[14] Jia Wang. A survey of web caching schemes for the internet. *SIGCOMM Comput. Commun. Rev.*, 29(5):36–46, 1999.

6 Appendix

6.1 Proof of Lemma 1

In [12], the following alternative interpretation of the request generation model is provided: "At each time slot, toss an $(h+1)$-sided, biased coin to decide the value of R_n. Let V_n be a random variable indicating the outcome of the toss at time $n > h$. Then, $P(V_n = i) = w_i$ for $1 \leq i \leq h$ and $P(V_n = h+1) = \beta$. Using this notation, the model is described by the following equation: $R_n =$

$\sum_{i=1}^{h} R_{n-i} 1_{\{V_n=i\}} + Y_n 1_{\{V_n=h+1\}}$." Thus,

$$c_j(i) = P(R_{n+i} = j | R_n = j) = P(R_n = R_{n-i} | R_{n-i} = j)$$
$$= P(V_n = h+1, Y_n = j) + \sum_{k=1}^{h} P(V_n = k, R_{n-k} = R_{n-i} | R_{n-i} = j)$$
$$= \beta q_j + w_i + \sum_{k=1, k \neq i}^{h} w_k c_j(|i-k|). \tag{25}$$

6.2 Proof of Lemma 2

It can be easily concluded that

$$H_j(k) \leq \widehat{H}_j(k), \forall k \geq 1. \tag{26}$$

More specifically, $H_j(k) = \widehat{H}_j(k), \forall k > h$ by definition, and, for $k \leq h$, (26) is derived directly by observing equations (15) and (16), since it holds that $c_j(k,m) \leq c_j(0,m) = c_j(m)$ (by definition).

The proof is made by induction on x. By definition, $H_j(1) = \widehat{H}_j(1)$ and thus the Lemma holds for $x = 1$, since $F_j(1) = G_j(1) = H_j(1) = \widehat{H}_j(1) = \widehat{G}_j(1) = \widehat{F}_j(1)$. Now assume that the Lemma holds for $x - 1$, that is assume that $F_j(x-1) \leq \widehat{F}_j(x-1)$. It must be shown that $F_j(x) \leq \widehat{F}_j(x)$. By definition[8],

$$\widehat{F}_j(x) = \widehat{F}_j(x-1) + (1 - \widehat{F}_j(x-1))\widehat{H}_j(x), \tag{27}$$
$$F_j(x) = F_j(x-1) + (1 - F_j(x-1))H_j(x), \tag{28}$$

[8] Based on equations (9) and (11), it holds that $F_j(x) = F_j(x-1) + (1 - F_j(x-1))H_j(x)$.

and, thus,

$$\widehat{F}_j(x) - F_j(x) = (\widehat{F}_j(x-1) - F_j(x-1))(1 - \widehat{H}_j(x))$$
$$+(1 - F_j(x-1))(\widehat{H}_j(x) - H_j(x)) \geq 0.$$

6.3 Derivation of the hit ratio for $h = 1$

In the special case where $h = 1$, $H_j(k)$ (see equation (15)) is given by

$$H_j(k) = \begin{cases} w_1 + \beta q_j, & k = 1; \\ \\ \beta q_j, & k > 1, \end{cases}$$

where $w_1 + \beta = 1$. Since it holds that $F_j(x) = F_j(x-1) + (1 - F_j(x-1))H_j(x)$ and $F_j(1) = G_j(1) = H_j(1) = w_1 + \beta q_j = 1 - \beta(1 - q_j)$, it can be easily obtained that $F_j(x) = 1 - (1 - \beta q_j)^{x-1}\beta(1 - q_j), x \geq 1$ and (from equation (8)) $F(x) = \sum_{j=1}^{N} q_j - \sum_{j=1}^{N}(1 - \beta q_j)^{x-1}\beta q_j(1 - q_j) = 1 - \sum_{j=1}^{N}(1 - \beta q_j)^{x-1}\beta q_j(1 - q_j)$, $x \geq 1$. Thus, for $h = 1$, equations (6) and (7) may be written as

$$C \approx s(T) = \sum_{z=0}^{T-1}(1 - F(z)) = 1 + \sum_{z=1}^{T-1}\sum_{j=1}^{N}(1 - \beta q_j)^{z-1}\beta q_j(1 - q_j)$$
$$= 1 + \sum_{j=1}^{N}\sum_{z=1}^{T-1}(1 - \beta q_j)^{z-1}\beta q_j(1 - q_j) = 1 + \sum_{j=1}^{N}(1 - q_j)(1 - (1 - \beta q_j)^{T-1})$$
$$= N - \sum_{j=1}^{N}(1 - q_j)(1 - \beta q_j)^{T-1}, \tag{29}$$
$$HR \approx F(T) = 1 - \sum_{j=1}^{N}(1 - \beta q_j)^{T-1}\beta q_j(1 - q_j). \tag{30}$$