Two New Metrics to Evaluate the Performance of a Web Cache with Admission Control

F.J. González-Cañete, E. Casilari, A. Triviño-Cabrera
Departamento de Tecnología Electrónica
University of Málaga
Málaga, Spain
{fgc, ecasilari, atc}@uma.es

***Abstract*—This work presents a new metric that calculates the performance of a Web cache with admission control. This metric is an alternative to the classic Hit Rate (HR) and Byte Hit Rate (BHR) that has been especially developed to measure the effectiveness of the cache when there is an admission control policy that decides if a new document that reaches the cache must be cached. Additionally, a second metric that computes the capacity of the admission control to reject documents that are referenced only one time or are modified from reference to reference, has been proposed. We have evaluated the ability of the metrics by means of simulation of a cache that considers the Least Recently Used (LRU) replacement policy and an admission control that rejects the documents based on their size.**

I. INTRODUCTION

The main function of a Web proxy cache is to store the documents that are requested by the users so that the next time these documents are requested, they are served from the cache instead of the original server. The Web proxy caches are usually located near the clients and hence the Web response time that users perceive is decreased as well as the traffic that the Web servers receive.

Once the cache storage space in the proxy is full and a new document has to be allocated in the cache, a replacement policy algorithm must be performed to decide which documents will be evicted from the cache to make room for the new one. The objective of the replacement policy is to evict the documents with the lowest probability of being referenced again. Caching researchers have proposed numerous replacement policies that can be grouped into three categories based on the criteria utilized to select the documents to be removed: recency-based, frequency-based and size-based [1]. The recency-based algorithms are based on the supposition that the documents that are requested are going to be requested again soon, so that the documents that must be evicted are those that were referenced for the longest time. As a recency-based algorithm, the LRU (Least Recently Used) algorithm was proposed to be used in main memory caching although actually it is the most widely implemented in Web caches such as Squid [2]. The LFU (Least Frequently Used) is a frequency-based replacement policy that removes from the cache the documents with the smallest access count. On the other hand, the size-based algorithms suppose that big documents are less accessed than small ones, so that the big documents are more probably evicted than the small ones. The LFF (Largest File First) is an example of replacement policy that evicts the largest file in the cache to make room for the new one. Other algorithms such as GDS (Greedy-Dual Size) [3] are recency-based and size-based simultaneously, while GDSF (Greedy-Dual Size with Frequency) [4] and GD* (Greedy-Dual*) [5] can fit in the three groups as they combine the three policies.

In spite of the numerous replacement policies developed, only a few algorithms have been proposed for the cache admission control, i.e. the algorithms that must decide if a document should enter the cache or not with independence of the cache status. Abrams et al [6] proposed a variation of the LRU replacement policy called LRU-Threshold that does not cache documents larger than a threshold size. This heuristic has the drawback that the optimal value of the threshold depends on the workload and the cache size. To solve this problem, Markatos [7] proposed the LRU-Adaptive algorithm that is similar to the LRU-Threshold policy except for the fact that the threshold value is dynamically recalculated based on the evolution of the performance of the cache. Aggarwal et al [8] proposed an admission control policy that uses a small auxiliary LRU cache of meta-information of the documents such as the timestamps of the last access in terms of the number of accesses and time, access cost and expiration data. When a document reaches the cache the admission control checks if it is present in the auxiliary cache. If it is, it enters the main cache only if the heuristic value assigned to the document based on the metadata is greater than the value of the documents to be evicted. Otherwise, the document meta-data is added to the auxiliary cache.

The rest of this document is organized as follows. In section II a metric that estimates the performance of a cache

This project was partially supported by the public project Nº TEL2003-07953-C02-01

with an admission control policy and a second metric to measure the performance of the admission control are proposed. In Section III a workload is characterized and utilized to evaluate by means of simulations the performance metrics proposed. Finally Section IV enumerates the main conclusions and possible future work of this paper.

II. PERFORMANCE METRICS PROPOSED

A. Cache metrics

Two metrics are conventionally utilized to evaluate and compare the efficiency of the replacement policies:

- HR (Hit rate): It is defined as the total number of requests that cause a hit in the cache (i.e. the document is already present in the cache when the request is performed) divided by the total number of requests.
- BHR (Byte Hit Rate): It is defined as the summation of the document sizes that cause a hit in the cache divided by the size of the documents processed.

The HR is a metric of the proportion of documents that have been served from the cache instead of the original server and it gives an idea of the reduction of the latency observed by the users. On the other hand the BHR is a metric of the proportion of traffic that has been served by the cache instead of the original server and it shows the bandwidth saved by the cache.

Unfortunately, these metrics are not adequate if we use an admission control policy because they consider every document that is requested even if the admission control has decided not to cache it and it is not referenced again, i.e. the admission control policy was able to discard a document that would have wasted storage space in the cache without causing a hit.

To solve this drawback we define the NUHR (Not Unique Hit Rate) and NUBHR (Not Unique Byte Hit Rate) metrics as shown in (1) and (2) respectively:

$$NUHR = \frac{\#Hits}{\#Total_req - \#R_ok} \tag{1}$$

$$NUBHR = \frac{Size_Hits}{Size_Total_req - Size_R_ok} \tag{2}$$

Where *#Hits* is the total number of requests that cause a hit in the cache, *#Total_req* is the number of requests that enter the cache and *#R_ok* is the number of request that were correctly discarded by the control policy. Similarly, the terms of (2) are related to the size of the requested documents.

These metrics give us a more accurate approximation of the performance of the cache because they consider the effect of the admission control policy. If the cache does not implement an admission control policy, these metrics obtain the same results as the HR and the BHR.

The main drawback of the NUHR and NUBHR metrics is how to measure the number of documents that have been incorrectly discarded by the access control. The approximation utilized in this study is to consider that a document has been correctly discarded when it is not requested again in the workload or it has been modified since the last time it was requested, i.e. the document reference is unique or it is modified before it is referenced again.

To distinguish the modification of a document from the interruption of a transfer we compare the difference between the sizes of the successive requests to the same document. If the difference is less than 5% of the document size, we consider that the document has been modified and it has to be treated as a new document; otherwise we consider that a cancel has occurred [9].

B. Admission control metrics

The measurement of the admission control policy performance has been divided into two parts that measure the rate of correct rejections and the rate of correct acceptations respectively.

We define the ACHR (Access Control Hit Rate) and ACBHR (Access Control Byte Hit Rate) metrics as shown in (3) and (4) respectively:

$$ACHR = \sqrt{\frac{\#R_ok}{\#R_Total} \cdot \frac{\#Ac_ok}{\#Ac_Total}} \tag{3}$$

$$ACBHR = \sqrt{\frac{Size_R_ok}{Size_R_Total} \cdot \frac{Size_Ac_ok}{Size_Ac_Total}} \tag{4}$$

Where *#R_ok* is the number of requests that were correctly rejected, *#R_Total* is the number of rejections, *#Ac_ok* is the number of requests accepted that caused a hit in the cache and *#Ac_Total* is the number of requests accepted by the access control. Similarly, the terms of (4) are related to the size of the requests.

These metrics offer an idea of the goodness of the admission control policy measuring the correctness of acceptation and rejection of requests, so that a good admission control should maximize the number of correct acceptations (i.e. it accepts the requests that are going to cause a hit) and the number of correct rejections (i.e. it rejects the requests that are not going to cause a hit). As we commented for the UNHR metric we have considered that a document rejection is correct when it is not requested again in the workload because it is unique or the document is modified before it is referenced again. Further metrics should

contemplate that a document is properly rejected if the next request occurs after a certain period of time.

III. METRICS EVALUATION

A. Workload Characterization

The workload utilised in this work corresponds to the traffic of a Squid proxy cache located in the Research Triangle Park (North Carolina, USA) from the 7th to the 11th of June 2004.

This trace has been processed to purge those requests that have been dynamically generated by CGI (Common Gateway Interface) discarding the requests that contain the strings 'cgi', 'cgi-bin' or '?'. As cacheable response codes, 200 (OK), 203 (Partial), 206 (Partial Content), 300 (Multiple Choices), 301 (Moved) and 302 (Redirects), have been considered [10]. For the requests in the trace with a 304 (Not Modified) response code the document has been requested again to the original server to obtain the real size. Table I summarises the basic characteristics of the trace after its process.

The relationship between the size of the documents and the number of references is depicted in Fig. 1. As it can be observed, the documents are more accessed as the size decreases. We will use this characteristic to implement the access control policies utilized in the next section to evaluate the performance of the cache and the access control policy.

B. Performance evaluation

To process the workload and calculate the performance metrics proposed, a proxy cache with an admission control simulator has been developed. This simulator implements various replacement policies and some simple access control policies based on the size of the documents and can be configured to simulate different sizes of cache. The simulator processes the workload files returning a result file that contains parameters such as the HR and BHR, the total size of the cache, the number of documents evicted, the total number of documents at the end of the simulation, etc. 10% of the trace has been used to "warm up" the cache and avoid cold start influences.

The admission control policy utilized is not to cache those documents that are greater than a certain threshold size. The sizes selected are four multiples of ten of the median of the documents size. As a replacement policy, the classical LRU has been used.

The cache sizes considered are 2%, 5%, 10%, 30% and 50% of the size of a hypothetical infinite cache that stores all the documents of the workload.

Fig. 2 compares the HR and BHR with the UNHR and UNBHR for the four access control policies considered respectively. As can be observed the difference between the HR and the NUHR is insignificant except for the more restrictive threshold, however the differences are more

TABLE I. MAIN WORKLOAD CHARACTERISTICS

Number of Requests	4,040,036
Size (GB)	40.4
Distinct documents	1,713,903
One timers (documents)	1,299,217
Mean (Bytes)	10,006
Median (Bytes)	1,720
Standard Deviation (Bytes)	229,941

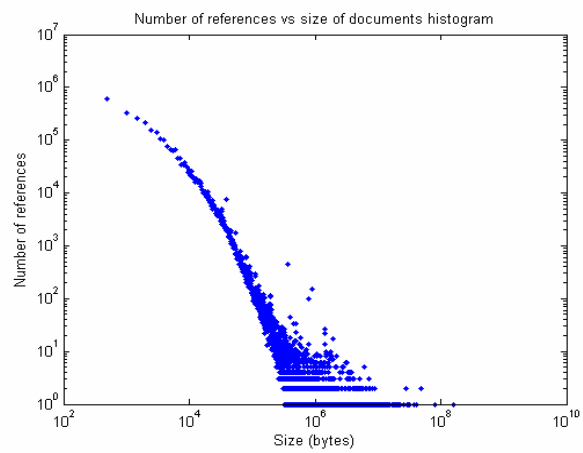

Figure 1. Histogram of the number of references as a function of the document sizes

evident between the BHR and the NUBHR, and this difference increases as we reduce the threshold size. The main drawback that was concluded in previous works was that rejecting documents largest of a certain size reduces drastically the BHR obtained. This is due to the fact that the BHR is an inadequate metric for a cache with admission control. The NUHR and NUBHR metrics proposed is then more accurate for this kind of cache and reveal that rejecting some documents does not obtain as bad performance as it was demonstrated.

Fig. 3 shows the ACHR and the ACBHR obtained for the admission control policies proposed previously. These figures evidence that the best option is to apply a threshold of 1,720,000 bytes to the size of the documents, especially for the ACBHR metric, although the ACHR obtained is similar to the threshold of 17,200 bytes. This effect is due to the fact that as can be observed in Fig. 1, a threshold size of 17,200,000 bytes only rejects a few documents that are referenced one time so the correct rejection ratio will decrease. Alternatively a threshold of 172,000 bytes or less rejects a lot of documents that are referenced more than one time and hence the correct acceptation ratio will decrease. Consequently, the metrics proposed offers a compensated measurement of the two main purposes of an admission control policy: to maximize the correct rejection ratio and to maximize the correct acceptance ratio.

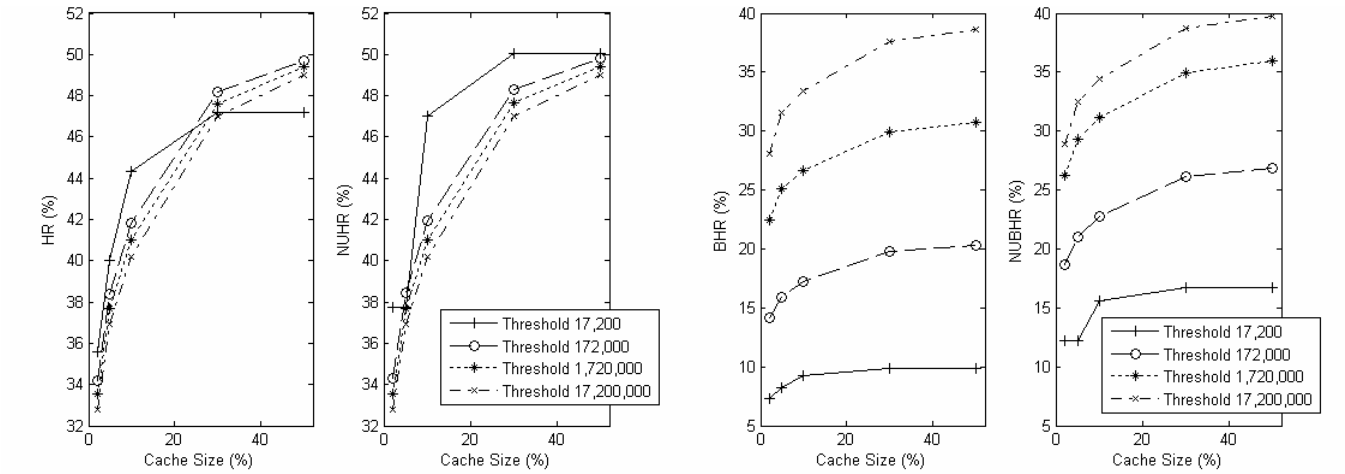

Figure 2. Comparation of the HR and NUHR (left) and the BHR and NUBHR (right) measured for the four admission control policies

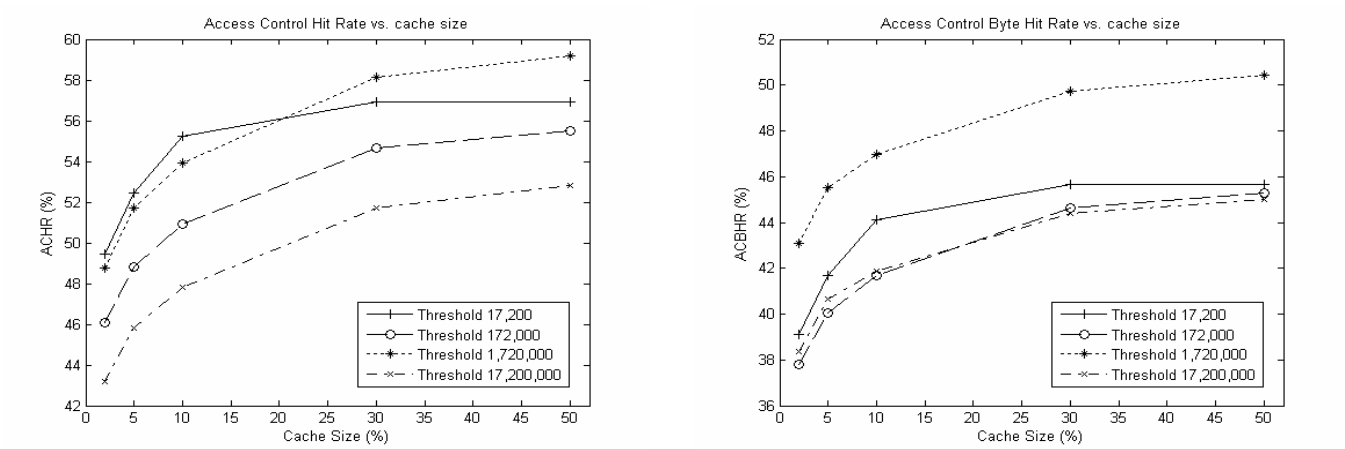

Figure 3. ACHR (left) and ACBHR (right) measured for the four admission control policies

IV. CONCLUSIONS

In this work two new metrics that determine the performance of a cache with an admission control and the admission control performance respectively have been proposed. By means of simulations, four admission control policies for a LRU cache have been evaluated demonstrating the goodness of the metrics proposed.

As a future work we propose to refine the measurement of the documents that are considered as incorrectly rejected by the admission control taking into account the probability of a document to be evicted from the cache before it will be referenced again.

REFERENCES

[1] R.A. Khayari, "Workload-Driven Design and Evaluation of Web-Based Systems", Der Andere Verlag, 2003.

[2] http://www.squid-cache.org

[3] P. Cao, "Cost-Aware WWW Proxy Caching Algorithms", Proceedings USENIX Symposium on Internet Technologies and Systems, Monterey, CA, USA, 1997.

[4] L. Cherkasova, "Improving WWW Proxies Performance with Greedy-Dual-Size Frequency Caching Policy", Technical Report HP Labs HPL-98-69, 1998.

[5] S. Jin, A. Bestabros, "GreedyDual* Web Caching Algorithm: Exploiting the Two Sources of Temporal Locality in Web Request Streams", Intl' Journal of Computer Communications, Vol. 24, No. 2, pp. 174-183, February, 2001.

[6] Abrams, M., Stanbridge, C.R., Abdulla, G., Williams, S., Fox, E.A., "Caching Proxies: Limitations and Potentials", Proceedings of the WWW-4 Boston Conference, December, 1995.

[7] E.P. Markatos, "Main Memory Caching of Web Documents", Proceedings of the fifth international World Wide Web conference on Computer networks and ISDN systems, Paris, 1996.

[8] C. Aggarwal, J.L. Wolf, P.S Yu, "Caching on the World Wide Web", IEEE Transactions on Knowledge and Data Engineering, Vol. 11, nº 1, pp. 94-107, Jan/Feb, 1999.

[9] M. Arlitt, R. Friedrich, T. Jin. "Workload Characterization of a Web Proxy in a Cable Modem Environment", Technical Report HPL-1999-48, Hewlet-Packard Laboratories, 1999.

[10] X. Zhang , "Cachability of Web Objects", Technical Report 2000-19, 2000.