A Game Theoretic Approach to Web Caching

S.Hadjiefthymiades, Y.Georgiadis, L.Merakos

Communication Networks Laboratory, Department of Informatics and Telecommunications,
University of Athens, Panepistimioupolis, Ilisia, Athens 15784, Greece
shadj@di.uoa.gr

Abstract. In this paper, the Game Theoretic framework is applied to Web caching. The interaction of multiple clients with a caching subsystem is viewed as a non-cooperative game. Some clients may continuously request resources, occupy a large segment of the cache disk space and thus, enjoy high hit rates. Owing to this situation, the remaining clients may suffer the removal of their "important" resources from the cache, and, subsequently, experience numerous cache misses. A utility function is introduced and calculated by clients in a decentralized fashion to avoid such monopolizing scenarios and guarantee similar performance levels for all users.

1 Introduction

Since the early '90s Game Theory has been extensively used in networking problems. Seminal papers like [3] and [4] provide a very insightful consideration of problems like bandwidth allocation and datagram switch operation disciplines. In this paper, we discuss the application of Game Theory in Web caching. The breathtaking increase in the volume of Web content world-wide renders the caching of resources a very important and promising area of research. Web caching has been extensively used to expedite users' queries by shortening the request-response chain. In this paper, our objective is to avoid having a single client monopolizing the allocated disk space in the Web caching proxy and thus, achieve high hit rates in contrast to the performance achieved by other users. As discussed in [10], the performance (hit rate) achieved in known Web cache servers is characterized by increased variance; in some cases, the variance in hit rates exceeds the average hit rate. Such statistical evidence clearly indicates a very wide range of performance levels seen by the users of the caching service. The hit rate variance is a decreasing function of the number of requests but persists even at high numbers of requests. To avoid such situations, we introduce a game theoretic mechanism that takes into account the disk space already allocated to a specific client and the actual benefit obtained by the retrieval of resources. Our scheme is based on a concave utility function, which secures the existence of Nash equilibrium points (NEP), in the considered game. The paper is structured as follows. In Section 2, we elaborate on the details of the utility function. Section 3 discusses the simulation set-up that we have adopted for evaluating the performance of the suggested solution and the respective results. Other Game Theoretic

studies of networking problems are discussed in Section 4. Section 5 concludes the paper.

2 Problem Statement and Game Theoretic Solution

As discussed above, our objective is to avoid having a single client monopolizing the disk space in the Web cache and thus, achieve high hit rates in contrast to the performance achieved by other users. Continuous requests by some clients will cause the reservation of a constantly increasing disk segment, force out the popular resources of other users and enjoy high hit rates. Users that do not interact in such a systematic way with the cache, suffer very low hit rates, cache misses and high response times. To cope with the problem, we introduce a utility function that takes into account the disk space already allocated to the specific client and the actual benefit obtained by the retrieval of resources. Such utility function is structured so as to secure the existence of NEP in the considered game. The interacting client calculates the value of the utility function (U) based on feedback received by the proxy cache (piggybacked in the HTTP responses delivered by the proxy). Whenever the marginal utility (ΔU) drops below a certain positive threshold (Eq.1), the cost for the client for the specific resource retrieval increases very rapidly and the client instructs the proxy to cease caching the retrieved objects.

$$\Delta U = U_{t+\Delta t} - U_t < \varepsilon,\ \varepsilon > 0,\ \Delta t > 0 \tag{1}$$

Hence, the disk space allocated to the user rests below a threshold and a stable convergence is secured. The utility function U_k for user k has as follows:

$$\begin{aligned} U_k &= P_k - C_k \qquad P_k = \frac{\alpha \cdot P_{hit}(k) \cdot R_k \cdot S}{[1 - P_{hit}(k)] \cdot R_k \cdot S} = \frac{\alpha \cdot P_{hit}(k)}{1 - P_{hit}(k)} \\ C_k &= \frac{P_{hit}(k) \cdot R_k \cdot S}{CacheSize - \sum_i P_{hit}(i) \cdot R_i \cdot S} = \frac{P_{hit}(k) \cdot R_k \cdot S}{CacheFreeSpace} \end{aligned} \tag{2}$$

The term $P_{hit}(k)$ denotes the caching performance achieved by user k. The term α is a normalizing constant (in our simulations, α=1.1). Term R_k denotes the number of requests made by user k towards the caching system. S is the average size of a Web resource file and can be used to determine the mean retrieval/storage cost associated with the resource. CacheSize denotes the total size of the cache disk space. The CacheFreeSpace term denotes the free cache disk capacity and, practically, represents the strategies of all involved players (i.e., shows the cache disk space reserved by the competing players). Term P_k represents the benefit (profit) owing to the caching capability of the proxy subsystem. The term C_k represents the cost owing to the retrieval of resources and the allocation of disk capacity. The term $P_{hit}(k)$ can be approximated by the Web client (agent) as a function of the number of requests issued by the client. Specifically, $P_{hit}(k)$ can be calculated by the R_k number of requests as follows [1].

$$P_{hit}(k) = \gamma \cdot \ln(R_k) - \delta \tag{3}$$

Constants γ and δ are non-negative real numbers. The concavity of the utility function (Eq. 2) guarantees the existence of a NEP. As proven in [5], an equilibrium point exists

for every concave n-person game. For a game with multiple players, say two, the payoff function of each player (i.e., $\varphi_1(x)$, $\varphi_2(x)$) is dependent upon the strategies of all players (i.e., the point $x=\{x_1, x_2\}$ belongs to the space S of feasible strategies which, in turn, is a subset of the Cartesian product $E_1 \times E_2$ of the domains of definition for strategy coordinates x_i). The function $\varphi_i(x)$ should be continuous in S and concave in x_i for fixed values of the other coordinate(s) (x_j, $j \neq i$). In the considered game, all payoff functions are identical, provided by (Eq.2). The strategy variable of user k is the number of requests issued by the user (R_k). The strategy variables of other users are taken into account in the formulation of the denominator of P_k. The strategy space E_i represents the number of requests issued by the Web user during interaction with the cache. Practically, this space is a bounded interval of $\aleph$, [0, u], where u is a fairly large number. Typically, u can increase as high as infinity. However, this is practically infeasible as Web sessions typically last 30 minutes. Additionally, an infinite increase of u would prevent us from applying the findings in [5] for establishing NEP existence. It can be easily shown that the utility function in (Eq.2) is concave. Since, the considered game is a two-players game with identical utility functions, the space of acceptable allocations is a convex, closed and bounded area in $\aleph^2$. The concave utility function guarantees the existence of a NEP.

Below, we examine the main characteristics of the utility function. The formulation of the utility function is based on the deduction of two terms. The first term (P_k) denotes the gain for the involved player from the caching interactions. It is an increasing function of the number of client requests. The second term (C_k) denotes the cost induced to the client through the retrieval and caching of resources. Such cost increases very rapidly as the remaining cache disk space approaches zero. It is known that P_{hit} can range to as high as 50% [8]. Analyses of extensive access traces from second and third level proxy caches, presented in [1], show that hit rates vary up to 45%. In [9] a maximum hit rate of 49% is reported for infinite cache size. Based on the above, the term P_k may increase from 0 up to the α constant. The structure of the P_k term favors the intensive user activity (the obtained benefit is higher when P_{hit} increases). Another interesting issue is how the term C_k varies as a function of R_k. Should the cache free space be very limited the C_k term demonstrates a very rapid increase. The utility function allows a “new” user (i.e., with a limited history of requests) to interact freely with the cache (i.e., cache all the fetched files). A “new” user tries to exploit the structure of P_k and increase his benefit through intensive behavior. Conversely, the interaction of a user with increased number of requests is restricted by the availability of cache disk space. Hence, a user tries to reach a balance between increased benefit and increased cost. As discussed above, the Web proxy returns to the interacting client (supported by an agent) the free disk space that is currently available (piggypacked in HTTP responses). Such information is exploited by the client to decide whether the object subsequently requested by the user should be cached or not (if not found in the cache). HTTP/1.1 provides the means for such, selective, caching through the cache-control header directive and a "no-store" value in specific. To calculate the value of the utility function, the interacting Web client needs also to estimate the average size of the retrieved resources (S in Eq.2). The client maintains an estimate of the average resource size through a low pass filter (Eq.4).

$$S_{new}=w \cdot S_{old}+(1-w) \cdot S_{resource} \qquad 0<w<1 \qquad (4)$$

The term *w* is a smoothing factor, $S_{resource}$ is the size of the recently retrieved resource while S is the estimate of the average resource size.

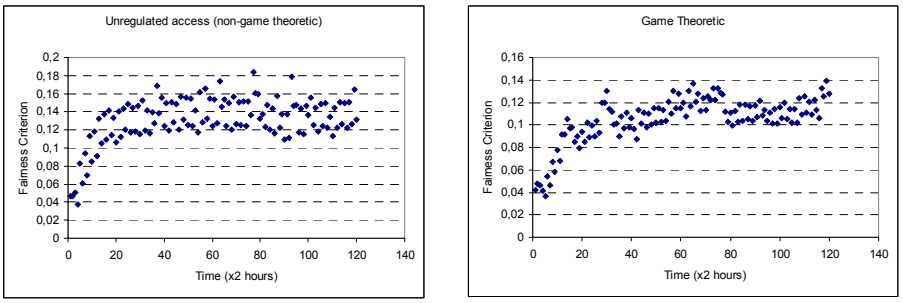
3 Simulation Framework and Results

We have to tried to assess the impact of the proposed game theoretic mechanism on a Web caching setting involving the intensive interaction of 1500 users (n) with a single cache for a period of 10 days. We have also simulated the non-game theoretic scenario. User interaction patterns followed the Web traffic model reported in [6] and [7]. The metrics recorded throughout the simulation were the number of requests made by each user (R), the number of observed cache hits (CH), the number of cache misses that were affected by the game theoretic mechanism (i.e., were not found in cache, were retrieved but not, subsequently, cached) (GT), the number of invocations to the LRU cache replacement mechanism, the percentage of available cache free space. In this paper, the cache hit rate (H) is calculated as follows.

Game theoretic scenario	Unregulated, non-game theoretic scenario
H=CH/(R+GT)	*H=CH/R*

Measurements were collected for all simulated user objects every 120 minutes. We have adopted the coefficient of variation of the H_i measurements as an indicator of the fairness achieved by the caching scheme. A high value of this fairness criterion (FC) means that different users do not enjoy the same benefits from caching and some monopolize the disk space. On the contrary, a low FC value implies that the behaviour experienced by the majority of users is almost identical and all have been allocated an almost equal disk share. Specifically, the fairness metric is defined as follows.

$$FC = s/\overline{H} = \left[\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(H_i - \overline{H})^2}\right] \cdot \left[\frac{1}{n}\sum_{i=1}^{n} H_i\right]^{-1} \quad (5)$$



(a) (b)

Fig. 1. Fairness Criterion

As shown in Fig.1, the game theoretic mechanism achieves (a) lower values for the coefficient of variation of H (i.e., the fairness criterion) and (b) a more predictable caching behaviour since the plot in Fig.1.b is much more condensed than in Fig.1.a. Quite similar observations have been made for the standard deviation of H. The proposed mechanism

managed to drastically reduce the number of LRU (Least Recently Used) replacements in the Web cache. Specifically, the number of replacements in the game theoretic scenario was reduced to the 1.17% of the unregulated case. The LRU scheme removed the least recently used items of the cache to free the 15% of the allocated disk space. We have also observed how the cache completeness (or, reversely, the free cache space) varies as time progresses. The game theoretic solution achieves completeness levels between 90% and 99%, in contrast to the non-game theoretic scenario where completeness levels are uniformly distributed between 85% and 99%. The observed number of LRU replacements indicates that the disk space usage remains close to 85% for much more time in the unregulated scenario. If the allocated disk space is the resource that users are charged for, then the game theoretic case entails more total revenue to the cache operator. Lastly, it is important to assess the penalty that users have to pay for the de-monopolizing policy enforced by the game theoretic mechanism. The game theoretic solution achieves an average cache hit rate of 17-17,5 % (Fig.2). The unregulated ("laissez-faire") solution achieves higher cache hit rates, in the order of 21-22%. However, the game theoretic solution appears much more predictable, since all the relevant points lie very close to each other (Fig.2). During our simulations, a very important observation was that equilibrium was reached very rapidly since the behaviour of all clients was governed by the same traffic model. To monitor the performance of the suggested solution over an extended time period, a 20% of the population of users were initialised every 2 hours (i.e., the number of requests of each client were reduced to a very low level). It is implied, that this part of the client population are "new" users that are allowed to interact freely with the caching system.

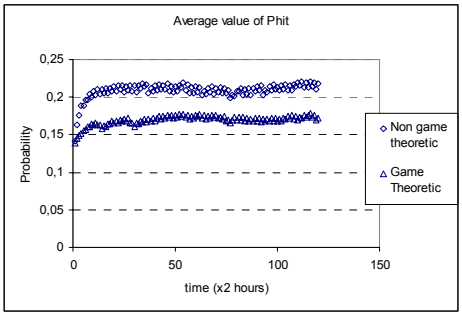


Fig. 2. Cache Hit Rate

4 Prior work

Game Theory has been extensively employed for resolving networking problems (e.g., flow control, routing). Nevertheless, to our knowledge, Game Theory has not been applied in Web systems engineering. In [2], the authors demonstrate that despite the non-cooperative decisions of network users, there is, still, room for network performance improvement. The authors in [3] discuss the available bandwidth distribution to ATM virtual paths controlled by different selfish users. The work in [4] focuses on the proper

design of the disciplines followed by network switches that could drive the network system to optimal conditions despite the selfish nature of the involved users.

5 Conclusions

In the context of Web caching, clients typically reserve more disk space in order to improve the observed cache performance. Such performance is denoted by the cache hit rate. As the cache disk space is a finite resource, a social interaction problem is formulated. Study of this problem is based on game theory. Specifically, we adopt the study of a non-cooperative game where the existence of a NEP is investigated. Users compete with each other trying to selfishly improve a utility function until a NEP is reached. We propose a utility function consisting of profit and cost components. Users has to determine a rational course of interaction taking into account the performance advantages and associated resource retrieval/storage costs. The cost component is dependent upon the strategies that different players assume. An extensive simulation of the game theoretic mechanism has been conducted. Our findings indicate considerable improvement in the adopted fairness criterion metric. The performance seen by different users is comparable and more predictable. At the game theoretic scenario, the number of cache replacement operations is drastically reduced and the cache enjoys higher utilization. The average hit rate seen by users is degraded at the game theoretic scenario and that is the penalty users have to pay for the de-monopolizing policy of operation.

References

1. B.M.Duska, D.Marwood, and M.J.Feeley, "The Measured Access Characteristics of WWW Client Proxy Caches", proceedings of USENIX Symposium on Internet Technologies and Systems, December 1997.
2. Y.Korilis, A.Lazar, and A.Orda, “Architecting Noncooperative Networks” IEEE JSAC, Vol. 13, No. 8, 1995.
3. A.Lazar, A.Orda and D.Pendarakis, “Virtual Path Bandwidth Allocation in Multiuser Networks”, IEEE/ACM Transactions on Networking, Vol. 5, No. 6, December 1997.
4. S.J.Shenker, “Making Greed Work in Networks: A Game Theoretic Analysis of Switch Service Disciplines”, IEEE/ACM Trans. Networking, Vol.3, No.6, December 1995.
5. J.B.Rosen, "Existence and Uniqueness of Equilibrium Points for Concave N-Person Games," Econometrica, Vol.33, No.3, 1965.
6. M.Crovella, and A.Bestavros, “Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes,” IEEE/ACM Trans. Networking, Vol. 5, No. 6, December 1997.
7. P.Barford, and M.Crovella, “Generating Representative Web Workloads for Network and Server Performance Evaluation,” Proceedings of ACM SIGMETRICS, July 1998.
8. E. Markatos, and C. E. Chronaki, “A Top-10 Approach to Pre-fetching the web”, Proceedings of INET ’98 Geneva, Switzerland, July 1999.
9. L.Fan, P.Cao, J.Almeida, and A.Z.Broder, “Summary Cache: A Scalable Wide-Area Web Cache Sharing Protocol”, IEEE/ACM Trans. Networking, Vol.8, No.3, June 2000.
10. C.Roadknight and I.Marshall, “Variations in cache behaviour”, in proceedings of 7th International WWW Conference (WWW7), Brisbane, Australia, April 1998.