Proceedings of the 1997 Winter Simulation Conference
ed. S. Andradóttir, K. J. Healy, D. H. Withers, and B. L. Nelson

MODEL-DRIVEN SIMULATION OF WORLD-WIDE-WEB CACHE POLICIES

Ying Shi
Edward Watson
Ye-sho Chen

Department of Information Systems and Decision Sciences
E. J. Ourso College of Business Administration
Louisiana State University
Baton Rouge, LA 70803, U.S.A.

ABSTRACT

The World Wide Web (WWW) has experienced a dramatic increase in popularity since 1993. Many reports indicate that its growth will continue at an exponential rate. This growth has created a tremendous increase in network loads and user response times. The complexity and diversity of many WWW documents (e.g., texts, images, video, audio, etc.) and the diversity of user requested WWW information require sophisticated WWW cache management strategies. Several popular WWW cache algorithms perform rather poorly and lack mathematical or empirical foundations. As a result, WWW system administrators and browser users are forced to arbitrarily define certain important cache parameters. Typically, such systems perform sub-optimally averaging hit rates below 55%. Our objective in this study is to develop a cache management strategy that is based on sound theory and principles from the information sciences and that can be utilized on-line, in real-time. Our approach is to study current cache algorithms and utilize actual empirical data to develop efficient and effective self-adaptive cache management strategies to handle anticipated Web growth.

1 INTRODUCTION

Increased user response times has left many web users cynically referring to WWW as the World-Wide-Wait (Abrams, 1997). Long wait times are attributed to an Internet bandwidth that has not increased at the same rate as the growth in demand being placed upon it. Bandwidth and response time problems will most likely increase as more people use the Internet. Thus, saving bandwidth, improving response time and reducing server load is a major research interest. The National Science Foundation states that a critical research topic for the National Information Infrastructure is to "*develop new technologies for organizing cache memories and other buffering schemes to alleviate memory and network latency and increase bandwidth*" (Bestavros 1995).

A cache is nothing more than a computer storage medium where certain documents are stored, often based on frequency and recency of document usage. The cache maintains a copy of certain documents from the origin servers to machines residing closer to clients. This can reduce the transmission distance significantly. Abrams (1995) states that without caching the WWW would become a victim of its own success.

Unlike a CPU cache, where a file is divided into many blocks that are homogeneous in size, a WWW cache contains documents of a widely varying size and type and the document is stored as a whole (Abrams 1995). Variable document sizes and types allow a rich variety of policies to select a document for removal, in contrast to policies for CPU caches that manage homogeneous documents (Williams, et al. 1996).

2. WEB CACHE PERFORMANCE, REMOVAL POLICIES AND USER ACCESS PATTERNS

2.1 Web Cache Performance

Several metrics are commonly used when evaluating Web caching policies. These include the following (Abrams, et al. 1997).

a) Hit rate - The hit rate is generally a percentage ratio of documents obtained through using the caching mechanism versus the total documents requested. In addition, if measurement focuses on byte transfer efficiency, weighted hit rate is a better performance measurement (Abrams, 1995).

b) Bandwidth Utilization - An efficiency metric. A reduction in the amount of bandwidth consumed shows the cache is better.

c) Response time/access time - The response time is the time it takes for a user to get a document.

There are various parameters such as user access patterns, cache removal policy, cache size and document size that can significantly affect cache performance.

Currently, the performance of several commercial Web caching software programs is not satisfactory. Typically, such systems perform sub-optimally, averaging hit rates below 55% (Pitkow and Recker 1994). Studies have shown that as many as half of the referenced documents are never referenced again. This would indicate that up to half the documents in a cache are useless, using a significant fraction of the cache and never yielding a cache hit.

2.2 Web Cache Removal Policies

Web cache removal policy can be classified into two different types in terms of the complexity: simple cache removal policies and hybrid cache removal policies.

2.2.1 Simple Cache Removal Polices

Simple and typical cache removal policies are FIFO (First-In-First-Out), LFU (Least Frequently Used), LRU (Least Recently Used). and SIZE (documents are removed according to document size). In this paper, we propose a simple Web cache removal policy: LWU (Least Weighted Usage) defined as follows:

$$\sum_{\tau=1}^{10} y_j(\tau)\gamma^{10-\tau}$$

where both frequency ($y_j(\tau)$) and recency (γ) are used to define this measure.

Pitkow and Recker (1994) in their studies of server caching found that ".. recency proves to be a stronger predictor than frequency." Their studies show that a mean hit rate of between 67% and 69% could be achieved by simply caching only the documents accessed one day ago. Wessel (1995) came to a similar conclusion for proxy caching, that one day is a key caching point. He found that 10 minutes is also an important caching point. Therefore, the results of Pitkow and Recker, and Wessel imply that LRU is better than LFU.

Abrams, et al. (1995) monitored traffic corresponding to three types of educational workloads over a one-semester period and used this as input to a cache experimental simulation. They found that when the cache is full and a document is replaced, the LRU removal policy performs poorly, but simple variations can dramatically improve hit rate and reduce cache size.

Williams, et al. (1996) propose a removal policy based on document size called SIZE. This SIZE removal policy suggests that when the cache is full, the largest document in the cache be removed. Through trace-driven simulation, they determined the maximum possible hit rate and weighted hit rate that a cache could ever achieve, and the removal policy that maximizes hit rate and weighted hit rate. The experiments use five traces of 37 to 185 days of client URL requests. The surprise is that removal policy based on SIZE always outperforms any other removal policies. They ranked hit rate performance of several removal policies in the following order (best first): SIZE, LRU-min, LRU, LFU.

2.2.2 Hybrid Cache Removal Policies

Hybrid cache removal policies are a combination of two or more simple removal policies. Some researchers indicate that hybrid cache removal policies would be more efficient than simple cache removal policies. Abrams, et al. (1995) argue that LRU ignores the document size. Thus to make room for, say, a large video for one user, it might replace many small documents resulting in misses for many users. They proposed two new algorithms of LRU variant: LRU-min and LRU-thold which try to minimize the number of documents replaced. The principle of LRU-min is to apply LRU only to the largest documents and then to groups of successively smaller documents. The LFU-aging policy, proposed by Arlitt, et al. (1997) is a variant of LFU. With simple LFU some documents can build up extremely high reference counts so that they are rarely removed. Occasionally, these high referenced documents may not be requested again.

2.3. User Access Patterns

Several researchers (Cunha 1995; Glassman 1994; Braun 1994,) have done web access analysis with simple methods such as web histograms and distributions. The analysis provides a picture of user access patterns that only represent Web access typical behavior rather than dynamic behavior. Web access phenomenon is in fact dynamic. A dynamic access modeling approach can provide insight into web access phenomena. Several researchers mention this issue (Pitkow and Recker 1996, 1994; Malpani 1996). Pitkow and Recker (1996, 1994) propose an algorithm that was derived from a psychological human memory model. This model focuses on frequency and recency of user access patterns. The cache hit rate can reach 67% for 2000 document accesses from off-campus for a three-month period. The advantage of this model is that it contains the dynamic property of user access pattern. But this model could not well reflect user access behaviors of frequency and recency. As they stated "… despite these findings, the relationship between recency and frequency in predicting document access remains unclear."

3. A DYNAMIC WEB USER ACCESS MODEL

Simon (1968) made a significant contribution to information processing with his research on word usage modeling and firm size growth modeling. We believe that there are similarities between word usage in text and Web access on the Internet. We apply Simon's model to Web access patterns. The purpose of this study is to develop a good model for user access patterns, and then to use a model-driven simulation approach to simulate Web caching policy.

3.1 Simon's Model - A Model with Autocorrelation

Simon formalized their second assumption as follows: Let $y_j(k)$ be the indicator of the access of the j-th document during the *k*-th time interval, where $y_j(k)$ is either one or zero with one meaning the document is accessed and zero meaning it is not accessed. Then the total number of accesses of the *j*-th document at the end of the k-th time interval is simply

$$\sum_{\tau=1}^{k} y_j(\tau)$$

The probability of accessing the *j*-th document during the (k+1)-st time interval can be formalized as:

$$p[y_j(k+1)=1] = \frac{1}{W_k}\sum_{\tau=1}^{k} y_j(\tau)\gamma^{k-\tau}$$

where W_k is the sum of weighted usage of all documents (which is a function of time k and is the same for all documents), and g is the parameter that determines how rapidly the influence of past access on new selection dies out.

3.2 Web Access Simulation and Analysis

Since it is observed that Web access has a Pareto distribution (Abrams, et al. 1995; Braun, et al. 1994; Cunha, et al. 1995; Markatos, et al. 1996), the purpose of this simulation is to determine if under the two assumptions of 'new entry' and 'autocorrelation', Web access data still has Pareto distribution.

To simulate Web access process, we collected two Web server logs: the Main Library of Louisiana State University (http://www.lib.lsu.edu/stats/usage.html), and the Core Resource Web Server at Columbia University (http://www.cis.columbia.edu/stats.html#Archive).

LSU access data covers all user accesses in 1996. Columbia University's data spans from April 12 to June 21 (1996). The total documents accessed at the web server is 509 with 156926 total accesses during this period.

3.2.1 The Estimation of Model Parameters

3.2.1.1 Estimating α and Estimating γ

Ijiri and Simon (1977) explain the technique for estimating α in their paper. The parameter α is the probability of a new entry. Thus the value of α is approximately equal to R/T, where R is the total number of different documents and T is the total number of usages. The ability to approximate the value of α from R/T is an intuitive relationship. It is known that α is the probability of the entry of a new document. Therefore, the total number of document accesses should be the product of T and α.

We estimate α for LSU data (R=2457, T=329385, and α = R/T = 0.0072594) and Columbia data (R=509, T=156926, and α = R/T = 0.003244). To estimate γ, we calculate areas of Pareto curves for different combinations of α and γ. Then, a linear regression equation is obtained as,

$$Area = -0.64622*\alpha + 0.341996*\gamma + 0.107573$$

Given α and the area calculated from empirical data, γ can be estimated with errors of the estimated γ roughly $\pm$ 0.03.

3.2.1.2 The Impact of Parameters on Pareto Curve

Figure 1 shows the significant role the new document entry rate α plays on shaping the Pareto curve. Specifically, the smaller the α, the more the graph curves to the northwest direction; and the larger the α, the more the graph curves to the y = x line. This is logical, because with a lower probability of new document entry, the accesses concentrate more on old documents. On the other hand, with higher α, the access spreads over more documents, thus less concentration is placed on old documents.

We know that a higher γ implies a slower rate of decay, i.e., a document's usage probability is little affected by it not being used. Thus, when γ = 1.0, there is no decay taking place at all. On the other hand, a small γ signifies that documents that have not been used recently will probably be neglected for a long to come regardless of how active they had been previously.

A related interpretation is that previously inactive documents do have a chance of becoming dominant in the future selection process even if this dominance is temporary. The curves in Figure 2 indicate that under most circumstances, this interpretation is plausible. However, when γ varies from 0.01 to 0.5 and α is held at 0.18, the curve shape is almost the same. This means that γ is not sensitive to usage concentration in the range of 0.01 to 0.5, when α is 0.18.

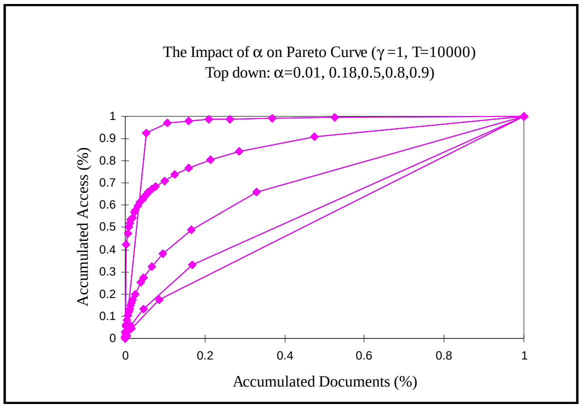

Figure 1 The Impact of α on Pareto Curve

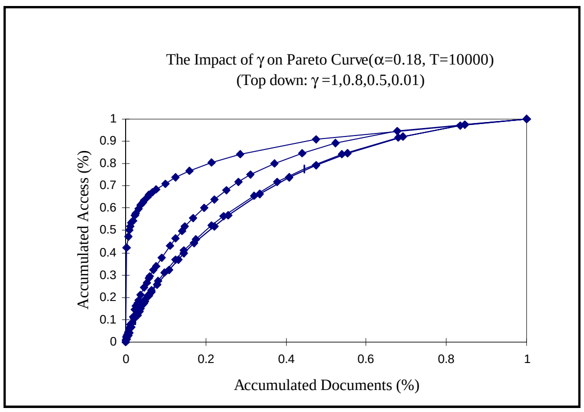

Figure 2 The Impact of γ on Pareto Curve

3.3.2 Simulation Results and Analysis

We use the LSU library web access data and Columbia University Web access data for comparison purposes. From LSU data, the total web access number in July 1996 is 329385 and the total documents are 2457. Therefore, α and γ are estimated using the above methods : α = 0.0074594 and γ =0.99. From Figure 3, we see that the curves fit well in the 'significant few' and 'trivial many' parts. The middle part data relate to the accuracy of estimated γ (Ijiri and Simon 1977) and is found to not match well. We believe that γ would be a function of web access τ, rather than a constant. However, the functional form is difficult to estimate.

Columbia University's web access data shows that total number of accesses is 156926 and total documents are 508. The parameters α and γ are estimated as 0.003244 and 0.96. Figure 4 suggests the same conclusions as above. Note that the LSU date are more concentrated (γ =0.99) than the Columbia data (γ =0.96). The entry rate of new documents at LSU is slightly higher than that of Columbia.

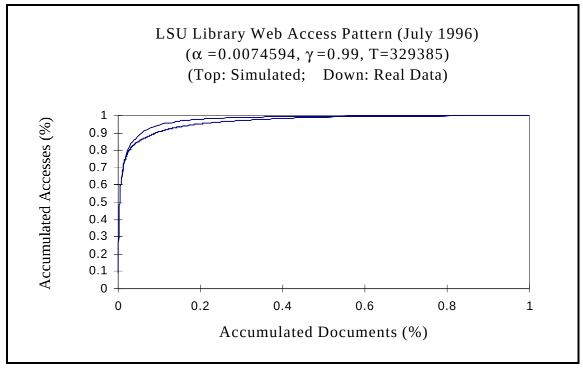

Figure 3 LSU Library Web Access Pattern

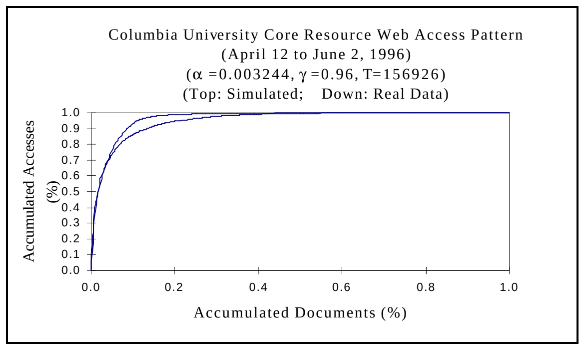

Figure 4 Columbia University Web Access Pattern

3.4 Summary

In this section, our objective is to justify the application of Simon's stochastic model to the Web access process. Below we present our primary findings.

a) Simon's approach provides a good benchmark for justifying a complex stochastic model.
b) Simon's model reflects user access behaviors of frequency and recency, and the characterization of Web document entry.
c) Web access data follows a Pareto distribution. The analysis of log data are consistent with past studies.
d) Based on two assumptions of Web accesses: 'new entry' and 'autocorrelation', we show that the generated data using Simon's model simulation still has the characterization of Pareto distribution.

e) In the middle of simulated Pareto curves, there is a mismatch. According to Simon's explanation, it probably stems from an inaccurate estimation of γ.

4 SIMULATION EXPERIMENTS

Abrams, et al. (1997) state that the user access model is an important consideration when designing cache schemes. The model attempts to characterize how the user will access Web documents. By knowing the user access pattern, better caching predictions can be made regarding what documents will be requested.

For simplicity, the assumptions of our simulation model of Web caching policies are as follows:

a) User requests are generated from a stochastic user access model - Simon model.
b) In this experimental simulation we ignore the simulation of document size and type.
c) Cache size is represented in the number of documents.
d) Caching removal policy is restricted to simple policies: FIFO, LFU, LRU, and LWU. Since document size is assumed to be a constant, SIZE removal policy is not simulated in the experiments.
e) The issues of cache staleness, private documents, and dynamic documents are not considered here.

4.1 Experimental Objectives and Design

4.1.1 Objectives

To investigate the impact of user access patterns on the Web cache removal policies we run experiments to answer the following questions:

a) How do frequency and recency affect the performance of the Web caching?
b) How does the entry rate of new Web documents affect the performance of the Web caching?
c) What is the best Web caching removal policy among FIFO, LFU, LRU and LWU?
d) How does cache size affect cache performance?

5.1.2 Experimental Design

In this experimental design, our objective is to investigate the impact of four factors on Web caching performance: α, γ, cache size, and cache removal policy.

We assign three levels (0.05, 0.1, 0.2) for α. We choose high entry rates for new documents to reduce the computational requirements of the experimental simulation. γ has two levels: 0.99 and 0.9999. Small γ indicates that user access patterns have preference on recency; large γ implies that user access patterns have preference on frequency.

In these experiments, we ignore the effect of document size on Web cache performance and assume all the document sizes are equal. Therefore, for simplification, cache size can be expressed as the number of documents. Similarly, since the documents are assumed of equal size, Weighted Hit Rate (WHR) is the same as Hit Rate (HR). Thus we use HR as the only measurement of cache performance.

Table 2: The Factors and Levels of Experiment

Factor	Level
α	0.05, 0.1, 0.2
γ	0.99, 0.9999
Cache Size (number of documents)	10,20,30, …, 290, 300
Caching Removal Policies	FIFO, LFU, LRU, LWU

Our experiments indicate that after 10000 user accesses, the hit rate is relatively stable. Thus the warm-up period of the simulation is specified as 10000 accesses in the experiments.

The scope of our experiments here focus on four cache removal policies: FIFO, LFU, LRU and LWU. In the future, we plan to extend our experimental scope to hybrid cache removal policies such as LFU-size.

4.2 Simulation Results

The simulation results of the Web cache performance are presented in Figure 5. The results are discussed in the sequence of factors: α, γ, cache size, caching removal policies.

4.2.1 The Results for α

These six graphs in Figure 5 show that in terms of cache hit rate, cache performance decreases when the entry rate of a new document (α) increases, regardless of whether γ is 0.99 or 0.9999. Intuitively, higher entry rates of new documents will dilute the concentration of user accesses from popular documents. Low concentrations of user accesses lead to a low cache hit rate.

Smaller α not only has a higher hit rate, it also converges faster (quickly reaches high hit rate) as it can be obviously seen from a, c, e in Figure 5.

4.2.2. The Results for γ

To study the impact of γ on cache performance, we compare graph a and graph b in Figure 5, in which γ varies from 0.99 to 0.9999 with fixed α (0.05). Two significant phenomena are observed from the graphs.

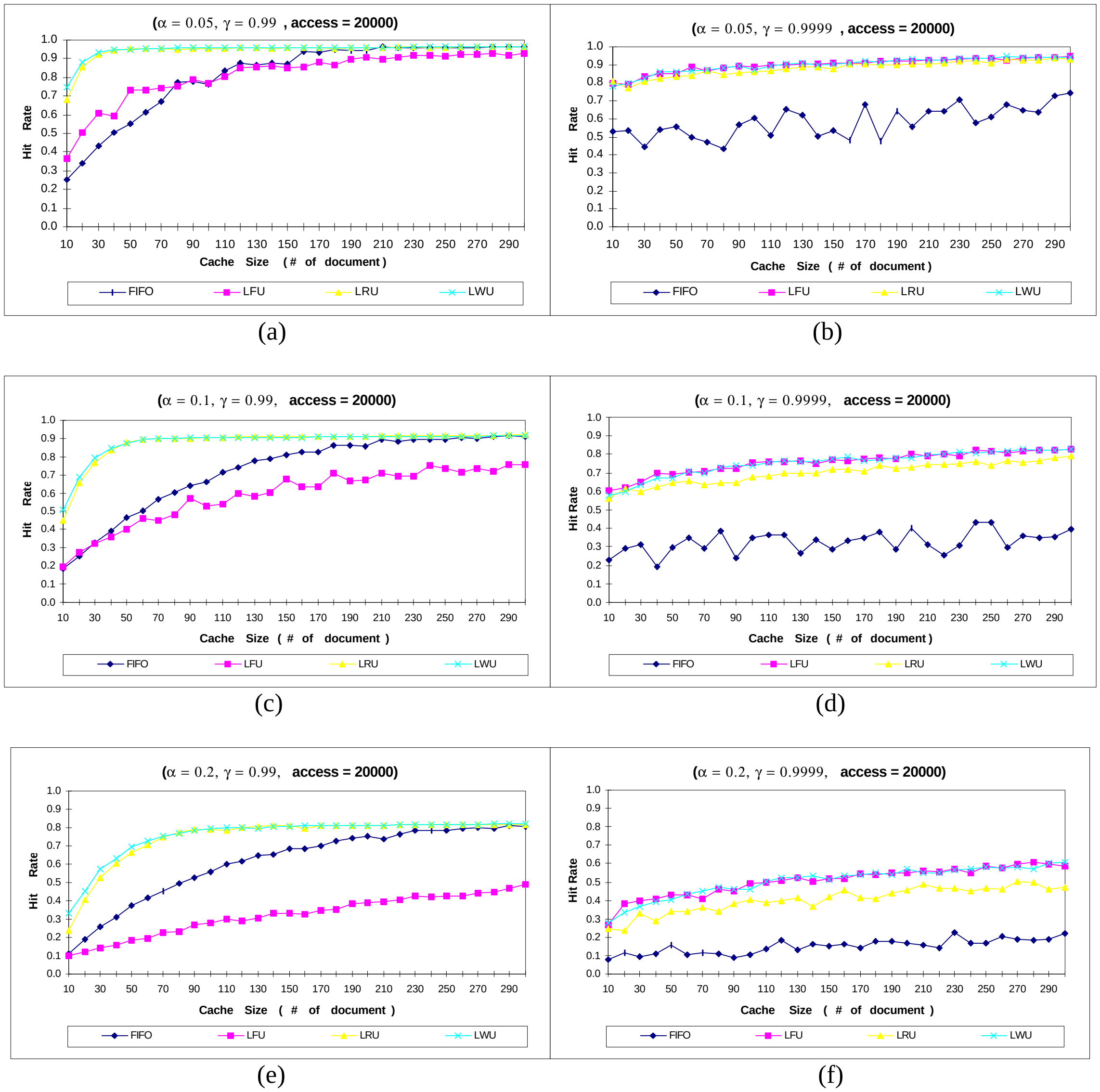

Figure 5: The Performance Comparison of Web Cache Removal Policies in Different User Access Patterns

a) The performance of the LFU removal policy is sensitive to γ. LFU performs relatively poorly when γ is 0.99. But LFU's performance largely improves as γ increases from 0.99 to 0.9999. In contrast, LRU's performance decreases slightly when γ increases from 0.99 to 0.9999. The sensitivity of LFU relative to γ is attributed to the fact that higher γ implies that the frequency preference of user access patterns.

b) Another observation from the two graphs is that the performance of FIFO, LRU, LWU worsens (except for LFU's performance) when γ increases from 0.99 to 0.9999. From Simon's model, we know that frequency preference of user access patterns (large γ) implies the low recency of user access patterns. Therefore, low recency (large γ) causes the poor performance of FIFO, LFU and LWU. In light of the controversy regarding LFU and LRU performance in the previous studies, we conclude that LRU performs well no matter what γ is. LFU has high performance only when γ is large.

This observation is consistent with the results of previous studies. Pitkow and Recker (1996) and Wessel (1995) indicate that recency is more important than frequency in terms of its impact on cache performance.

It is interesting to observe from all six graphs that if α is larger (0.1 or 0.2), the two significant phenomena discussed above are more obvious. This observation indicates that α and γ are interactive. Chen (1996) discusses this interaction. The interaction seems to be very complex.

4.2.3 The Results for Cache Size

Broadly speaking, larger caches get higher hit rates. This common knowledge is reflected in the graphs in Figure 5. However, it is very important to understand that beyond some cache size threshold, cache performance improves only marginally by adding more cache space. This phenomenon can be clearly observed from graph *a* in Figure 5. Also, the threshold varies with different removal policies and different α and γ values. The threshold is more obvious as γ gets smaller (see a, c, e in Figure 5).

The determination of the break point can save cache space in spite of a slight decrease in hit rate. This is very important for scarce cache resources in a proxy cache.

4.2.4 The Results for Cache Removal Policies

From Figure 5, we can observe the following phenomena regarding removal policies:

a) FIFO performs better than LFU when γ is 0.99, but it performs worst among the removal policies when γ is 0.9999.
b) LWU performs best no matter what α an γ are.
c) LFU and LRU performance is closely associated with γ, especially since LFU is more sensitive to γ. This phenomenon is discussed above.

FIFO reportedly performs poorly as a cache policy. Although most research in this area ignore this policy, it has never provided a reasonable interpretation of why it would perform poorly, or a confirmation of its poor performance using simulation. Through our simulation, the poor performance of FIFO is confirmed.

The excellent performance of LWU is expected, since it incorporates both frequency and recency, which are two important characteristics of user access patterns. If user access patterns change (i.e., α and γ change), LWU policy can be self-adapted to the change. Therefore, no matter what application environments are, LWU always capture users behaviors.

5. CONCLUSIONS

Through the simulation of Simon's model and the experimental simulation of Web cache policies, our conclusions can be summarized.

Simon's model is a robust dynamic model of Web user access patterns. It integrates frequency and recency into a simple mathematical model.

Model-driven simulation of Web cache policies has advantages over the trace-driven simulations most often used in research studies. Model-driven simulation allows a much broader range of access patterns for experimental purposes and allows us to link user access patterns with the performance of the Web cache policies.

LWU removal policys perform best due to their ability to adapt both frequency and recency in different application environments.

The performance of LFU and LRU removal policies has been the subject of heated debate. Through our experimental simulation we can see that this controversy is due in large part to the inconsistent results of various researchers that each assume specific (trace-driven) user access patterns from a unique site.

Our model-driven experimental simulation in this paper is currently being migrated from a spreadsheet to a simulation language to provide us with more modeling flexibility and faster execution speed. The investigation of document size, hybrid cache removal policies, and cache partitioning is ongoing.

REFERENCES

Abrams, M., et al. 1995. Caching Proxies: Limitations and Potentials, In *Proceedings of the Fourth International Conference on the World-Wide Web*, Boston.

Abrams, M., et al. 1997. WWW: Beyond the Basics (http://ei.cs.vt.edu/~wwwbtb/fall.96/book/chap25/index.html).

Arlitt, M, et al. 1997 (January). Trace-Driven Simulation of Document Caching Strategies for Internet Web Servers. *Simulation*: 23-33.

Bestavros, A. 1995. Demand-based Document Dissemination for the World-Wide Web. Technical Report BU-CS-95-003, Boston University Computer Science Department, Boston, MA.

Braun, H.W. and K. Clayffy. 1994 (May). Web Traffic Characterization: An Assessment of the Impact of Caching Documents from NCSA's Web Server. In *Proceedings of the Second International Conference on the World-Wide Web* (WWW94). Amsterdam, Elserver.

Chen,Y.S. 1997. Information Productivity Modeling: the Simon-Yule Approach. *Encyclopedia of Library and Information Science*, to appear.

Chen et al. 1994. Mathematical and Computer Modeling of the Pareto Principle. *Mathematical. Computer Modeling.* Vol. 19 (9): 61-80.

Cunha, et al. 1995. Characteristics of WWW Client-based Traces. Report, Department of Computer Science, Boston University (http://cs-www.bu.edu/faculty/crovella/paper-archive/TR-95-010/paper.html).

Glassman, S. 1994 (May). A Caching Relay for the World Web. In *Proceedings of the First International Conference on the World-Wide Web.* Amsterdam, Elserver.

Ijiri, Y. and H. A. Simon. 1977. *Skew Distributions and the Sizes of Business Firms*. North-Holland.

Luotonen, A. and Atlis, K. 1994 (May). World-Wide Web Proxies. In *Proceedings of the First International World Wide Web Conference.* Amsterdam, Elserver.

Malpani, R., J. Lorch, J., and D. Berger. Making World Wide Web Caching Servers Cooperate. *(http://www.w3.org/pub/conferences/www4/papers/59/)*

Markatos, et al. 1996 (May). Main Memory Caching of Web Documents. In *Proceedings of the Fifth International Conference on the World-Wide Web*. Paris.

Pitkow, J. E. and M. M. Recker. 1994 (May). A Simple Yet Robust Caching Algorithm Based on Dynamic Access Patterns. In *Proceedings of the First International Conference on the World-Wide Web.* Amsterdam, Elserver.

Recker, M. M. and E. J. Pikow. 1996. Predicting Document Access in Large, Multimedia Repositories. Report, Graphics, Visualization & Usability Center, Georgia Institute of Technology.

Simon, H. A. 1955. On a Class of Skew Distribution Function. Biometrika, Vol. 42: 425-440

Simon, H. A. and T. A. Van Wormer. 1963. Some Monte Carlo Estimates of the Yule Distribution. Behavior Science: 203-210

Wessels, D., 1995. Intelligent caching for World-Wide Web Objects. Master's Thesis, University of Colorado.

Williams, et al. 1996 (August). Removal Policies in Network caches for World-Wide Web Documents. In *Proceedings of The Association for Computing Machinery, Inc. (ACM) 96 Conference*. Stanford, CA.

AUTHOR BIOGRAPHIES

YING SHI is a Ph.D. student in the department of Information Systems and Decision Sciences at Louisiana State University. He received a BS degree in petroleum engineering from Daqing Petroleum Institute in 1985, and he received an MS degree in petroleum engineering from Louisiana State University in 1993. His current research interests focus on the application of stochastic modeling and simulation, system dynamics as well as World-Wide Web cache strategies.

EDWARD F. WATSON is an Assistant Professor in the Department of Information Systems and Decision Sciences at Louisiana State University. Previously he served on the consulting staff at Systems Modeling Corporation for 4 years. He received MS and PhD degrees in Industrial Engineering from Penn State. His interests include simulation modeling and analysis, supply chain management, business process reengineering, and system dynamics. He will serve as the Associate Editor for the *Proceedings of the 1998 Winter Simulation Conference*. Dr. Watson is a member of SCS, DSI, and INFORMS.

YE-SHO CHEN is an Associate Professor in the Department of Information Systems and Decision Sciences at Louisiana State University. He received a BS in mathematics from National Cheng Kung University in Taiwan, a MS in statistics from National Tseng Hua University in Taiwan, and a PhD in Industrial Engineering with an Operations Research Minor from Purdue University. His research interests and major publications are in data warehouse, information technology in small businesses, and information productivity modeling. Dr. Chen is a member of ACM, AIS, and INFORMS.