IEICE TRANS. COMMUN., VOL.E83–B, NO.7 JULY 2000

PAPER

Evaluation of Compulsory Miss Ratio for Address Cache and Replacement Policies for Restoring Packet Reachability

Masaki AIDA[†], ***Regular Member***, **Noriyuki TAKAHASHI**[††], ***Nonmember***, ***and*** **Michiyo MATSUDA**[†], ***Regular Member***

SUMMARY In high-speed data networks, it is important to execute high-speed address resolution for packets at a router. To accomplish high-speed address resolution, address cache is effective. For HTTP accesses, it has been discussed that the Dual Zipfian Model can describe the distribution of the destination IP addresses, and it enabled us to derive the cache miss ratio in the steady state, i.e., the cache miss ratio when the cache has full entries. However, at the time that systems are initialized or network topology is changed, the address cache has no address information or invalid address information. This paper shows the compulsory miss ratio which is the cache miss ratio when the cache has no address entry. In addition, we discuss the replacement policies of cache entries, for fast recovery of packet reachability, when the cache has information of unreachable address.

key words: *HTTP, cache, dual Zipfian model, compulsory miss, purge*

1. Introduction

To improve and enhance the IP-based Internet-type network, various network architectures has been proposed and studied, such as IP over ATM [1], NHRP [2], and so on. For the purpose of bringing a high speed, a rich bandwidth, and a wider area-cover to the connection-less network like the Internet, minimizing a delay in the packet processing has a much importance.

Packet processing in the routers within the connection-less network specifically includes the address resolution process, through which the router finds a link address, an ATM address for example, of the destination and/or the next-hop for the destination IP address of the packet. This process is indispensable for the packet transfer over physical links such as ATM, and efficiency of the address resolution process has significant influence to the throughput [3]–[5].

To speed up address resolution, edge routers which accommodates users are usually equipped with cache tables for the addresses. When the router has the cache entry for the destination of the packet (as the result of processing the earlier packet to the same destination), address resolution is immediately carried out. Otherwise, the router must utilize more exhaustive but timely-expensive method to resolve the address, such as requesting resolution to the address server [1] (Fig. 1(a)) and waiting for the response, or sending the packet to the default router [2] (Fig. 1(b)). Though either the address server or the default router has comprehensive address information, they can be a performance bottleneck if the frequency of cache miss in the network goes beyond their capacities. Therefore estimation of the cache miss ratio (or conversely the cache hit ratio) is important and desired as it has impact on the required performance of the address server or the default server, the network performance, and the design of the network.

The cache miss ratio is categorized according to situations as follows:

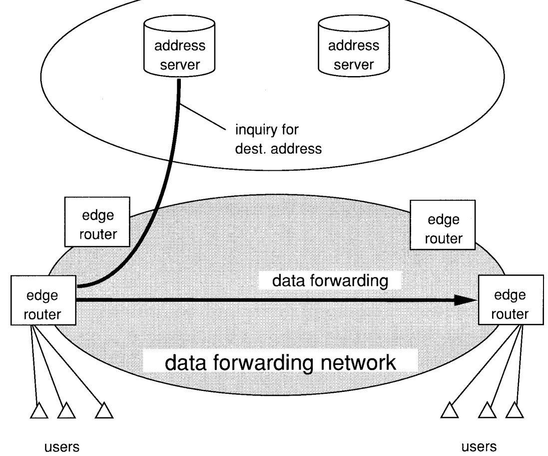
- Capacity miss ratio
  This is the cache miss ratio in the steady state, i.e., the cache miss ratio when the cache has full entries.
- Compulsory miss ratio
  This is the transitional behavior of the cache miss ratio when the cache is not full.

[4] discussed that the Dual Zipfian Model can describe the distribution of the destination IP addresses of HTTP traffic, and drive the capacity miss ratios in the steady state. Although system works in the steady state in usual, there are some exceptions. For example,

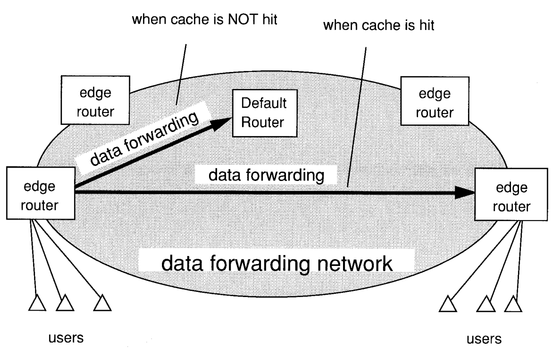
- At the initialization of a system:
  Re-initialization of a system is required when the system is in trouble. That time, all information in address cache will be lost.
- After purge operation of the address cache:
  When troubles on a transmission link occur or network topology changes, address information in the address cache can not guarantee packet reachability. In order to restore packet reachability, renewal of cache entries is required. A purge operation, which deletes all the cache entries, is used to restore packet reachability. After a purge operation, the address cache has no address information.

Manuscript received May 20, 1999.
Manuscript revised November 29, 1999.

[†]The authors are with Traffic Research Center, NTT Advanced Technology Corporation (NTT-AT), Musashino-shi, 180-0006 Japan.
[††]The author is with NTT Network Innovation Laboratories, Yokosuka-shi, 239-0847 Japan.



(a) Address resolution using an address server.



(b) Address resolution using the default router.

Fig. 1 Network models.

When the cache entries are lost after re-initialization or a purge, the cache miss ratio jumps high. After that, the cache miss ratio descends gradually and approaches the capacity miss ratio in the steady state. These transitional behaviors of the cache miss ratio are the compulsory miss ratio.

When the cache entries are lost and the cache miss ratio jumps high, it brings a serious deterioration of network performance. In case of Fig. 1(a), the router must request address resolution from the address server frequently, and the load of the address server greatly increases. In case of Fig. 1(b), the router sends the packet to its default router frequently, and the load of the default router greatly increases. These heavy loads are caused by the high compulsory miss ratio. In order to design actual networks and estimate their performance, it is important to evaluate the their compulsory miss ratio.

This paper shows the compulsory miss ratio when the cache has no address entry. In addition, by using both the compulsory and capacity miss ratios, we discuss the replacement policies of the cache entries, for fast recovery of packet reachability, when the cache has unreachable address information.

2. Background

2.1 Dual Zipfian Model

We consider the HTTP accesses from the users accommodated by the same edge router. Let the total number of HTTP accesses during a period be N, and the number of different destination IP addresses appeared in those N accesses be M. The relationship between N and M is represented by the following three equations:

$$M = H\left[\gamma + \frac{1}{\beta}\ln R + \frac{1}{2R^{1/\beta}}\right] + R^{(\beta-1)/\beta}, \quad (1)$$

$$N = R\left[\gamma + \frac{\beta-1}{\beta}\ln R + \frac{1}{2R^{(\beta-1)/\beta}}\right] + H R^{1/\beta}, \quad (2)$$

$$N = R\left[\gamma + \ln M + \frac{1}{2M}\right], \quad (3)$$

where R denotes the number of accesses to the most frequently accessed IP address, and H denotes the number of IP addresses which are accessed only once. In addition, γ denotes the Euler number ($\gamma \simeq 0.57721$), and β is a parameter ($2.2 \le \beta \le 2.5$) [4], [6]. The set of the above equations (1)–(3) is called the Dual Zipfian Model (DZM) [4]. For example, if N is given, we can obtain the estimations of M, R, and H using Eqs. (1)–(3).

DZM is applicable for not only HTTP accesses but also more realistic mixed packet stream which contains HTTP, FTP, and SMTP e-mail at an edge router [4]. In cases of a backbone router or a WWW server, DZM can be applied with some extension, too. We describe those extension shortly in Appendix A.

Hereafter, we shortly denotes the relationship between N and M represented by Eqs. (1)–(3), as

$$M = f(N). \quad (4)$$

2.2 Capacity Miss Ratio

Here, we review the derivations of the capacity miss ratio according to [4].

Let the capacity of the address cache (the maximum number of address entries) be k. When the cache does not contain the address, the edge router requests corresponding address information to the address server, and the address information is written over an entry of the address cache (Fig. 2). If all cache entries are occupied by address information, new address information is written over an existing entry determined by an aging algorithm.

Different aging algorithms give different capacity miss ratios, in general. In this paper, we consider the following two classical aging algorithms:

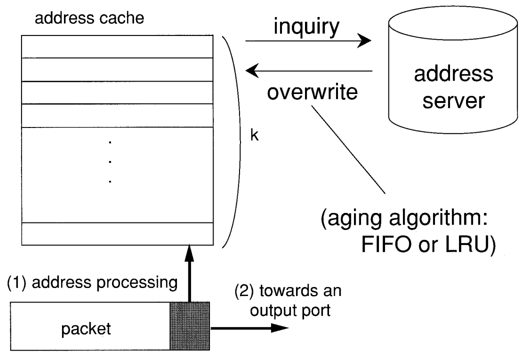


Fig. 2 Address cache model.

- Least Recently Used (LRU):
  When the cache does not hit, new address data is written over the least recently used entry.
- First In First Out (FIFO):
  When the cache does not hit, new address data is written over the current oldest entry.

For LRU, the address cache in the steady state always has k different IP addresses which are most recently accessed. The number of HTTP accesses that requires resolution of k different IP addresses is $f^{-1}(k)$. Similarly, the number of HTTP accesses that requires resolution of $k+1$ different IP addresses is $f^{-1}(k+1)$. Thus the number of HTTP accesses between two successive replacements of entries is $f^{-1}(k+1) - f^{-1}(k)$. The capacity miss ratio for LRU, P_L, is therefore obtained as

$$P_L = \frac{1}{f^{-1}(k+1) - f^{-1}(k)}. \tag{5}$$

For FIFO, unlike LRU, the address cache does not always have k different IP addresses that are most recently accessed. Instead, let N_x be the number of accesses in the period during that all cache entries are replaced. From [4], N_x must satisfy the following condition:

$$4\, f(N_x) - 3\, k - \sqrt{9\, k^2 - 8\, k\, \{f(2\, N_x) - f(N_x)\}} = 0. \tag{6}$$

If the capacity k is given, we can determine N_x by Eq. (6). Thus, the capacity miss ratio for FIFO, P_F, is obtained as:

$$P_F = \frac{k}{N_x}. \tag{7}$$

3. Compulsory Miss Ratio

Here, we investigate the compulsory miss ratio of the address cache, i.e., the relationship between the number of accesses and the cache miss ration after voiding all cache entries. In evaluation of the compulsory miss ratio, it is not necessary to take into consideration of the difference of aging algorithms, e.g., LRU or FIFO. The following discussion is applicable for any aging algorithms.

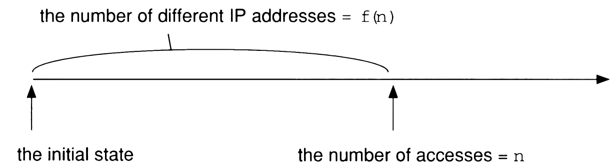


Fig. 3 Interval of cache misses.

At the initial state, the address cache (its capacity is k address entries) has no address entry. For each access, if the address cache does not hit, then the corresponding address data is written as an entry of the cache. When n accesses are happened from the initial state, the number of effective entries of the address cache is $f(n)$ by DZM (Fig. 3). Note that range of n is bounded by $f(n) \leq k$. The number of required accesses between the time of n-th access and the time of the next cache miss which brings a new valid cache entry is:

$$f^{-1}\left(f(n) + 1\right) - n. \tag{8}$$

Thus, the compulsory miss ratio, $p_c(n)$, at the time that n accesses occur from the initial state, is obtained as

$$p_c(n) = \frac{1}{f^{-1}\left(f(n) + 1\right) - n}. \tag{9}$$

The form of the compulsory miss ratio (9) is basically the same to that of the capacity miss ratio for LRU (5), since any cache contains most recently accessed addresses until the cache becomes full.

Next, we show the evaluation of the cumulative compulsory miss ratio $P_c(n)$. This is defined as the ratio of the number of cache misses to the number of accesses, from the initial state. The number of different IP addresses during n accesses is $f(n)$. This is equal to the total number of cache misses from the initial state. Therefore, the cumulative compulsory miss ratio is represented by

$$P_c(n) = \frac{f(n)}{n}. \tag{10}$$

Let us confirm the relation between Eqs. (9) and (10). We define a cumulative compulsory miss ratio between the $(n+1)$-th access and the m-th access $(n < m)$ as:

$$\delta_c(n, m) = \frac{f(m) - f(n)}{m - n}. \tag{11}$$

Equation (10) is actually a special case of Eq. (11) where $n = 0$, i.e.,

$$\delta_c(0, m) = P_c(m). \tag{12}$$

Furthermore, we can derive Eq. (9) from Eq. (11) by selecting m and n such that only one cache miss occurs between those accesses:

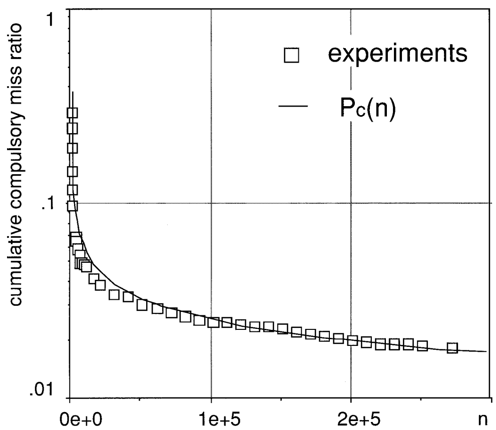


Fig. 4 Cumulative compulsory miss ratio (HTTP).

$$f(m) - f(n) = 1, \tag{13}$$

i.e.,

$$m = f^{-1}(1 + f(n)). \tag{14}$$

By substituting Eq. (14) for Eq. (11), we have Eq. (9).

4. Experimental Result

Although compulsory miss ratio (9) is an important performance measure in actual applications, it is difficult to compare it to experimental results, since the cache miss ratio changes instantaneously and is difficult to measure. In order to verify the validity of the compulsory miss ratio (9), we compare the cumulative compulsory miss ratio (10) with experimental results. If the validity of Eq. (10) is verified, Eq. (9) is also justified.

Figures 4 and 5 show the comparison between the estimated cumulative compulsory miss ratio (10) (denoted by solid lines) and experimental results (denoted by small square symbols). The vertical axis denotes the cumulative compulsory miss ratio in log scale, and the horizontal axis denotes the number of access from the initial state. There is no address entry at the initial state.

For Fig. 4, the experimental results come from HTTP access logs of a WWW proxy server in NTT Laboratories (about 100,000 accesses a day), and we calculate the actual cumulative compulsory miss ratio through trace-driven simulation using the logs. The value of a parameter β in DZM is 2.30.

Figure 5 shows the comparison for the mixed traffic which contains HTTP, FTP and SMTP e-mail accesses. Those accesses are also logged in NTT Laboratories. The value of a parameter β in DZM is 2.35.

From these figures, we can recognize the cumulative compulsory miss ratio (10) gives dependable evaluations.

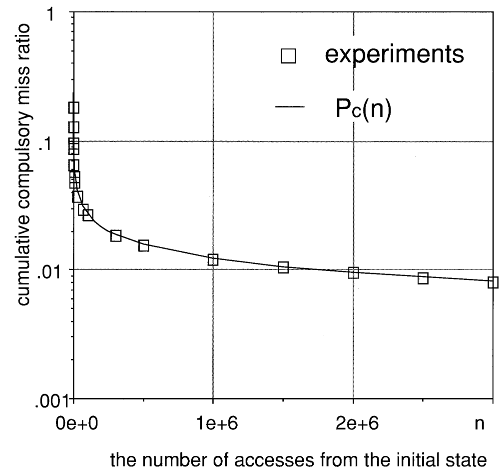


Fig. 5 Cumulative compulsory miss ratio (HTTP, FTP & SMTP e-mail).

Table 1 Comparison among schemes for restoring packet reachability.

	LRU	FIFO
Purge operation	good	moderate
Natural recovery	bad	good

5. Replacement Policies for Restoring Packet Reachability

When a link trouble or a topology change of the network occurs, address information in the address cache may lost its validity and cannot assure the packet reachability. In this situation, it is necessary to update the corresponding cache entries. In this paper, we discuss two schemes to update the invalid cache entries: full purge and natural recovery. Accordingly the number of possible combinations of aging algorithms (LRU and FIFO) and the strategies are four (Table 1).

5.1 Purge Operation for Address Cache

A full purge operation is initiated by the address server whenever topology changes. After full purge at an address cache, there are no address entries in the cache. The cache miss ratio jumps high instantly, then according with cache misses which actually increase the number of active cache entries, the cache miss ratio falls gradually. This behavior of cache miss ratio is described by the compulsory miss ratio (9). The address server usually accommodate multiple edge routers, where each of them has its own address cache. If the purge operations for multiple address caches are conducted simultaneously, the total amount of address resolution inquiry may exceed the capacity of the address server. To avoid this situation, it is necessary that multiple address caches get fully purged not at once but by turns. Interval between adjacent purge operations is determined by both the compulsory miss ratios of already purged caches and capacity miss ratios of the un-purged caches. Note that when multiple caches get purged by turns, the address server have to handle resolution requests from not only purged caches but also steady caches which wait their turns to be purged.

Note that there can be another scheme which purges invalid entries selectively (in contrast to full purge which purges all, valid and invalid, entries at once). This selective purge requires mechanisms to search the caches with invalid entries, and to search the invalid entries in each of caches. The selective purge is out of scope of this paper since those mechanisms are specific to network architecture protocols, and implementation.

Let us compare LRU and FIFO when purge operation scheme is used (Table 1). LRU is preferable to FIFO by the following two reasons:

- The capacity miss ratio for LRU is less than that for FIFO, and
- The compulsory miss ratios for both LRU and FIFO are the same.

A purge operation itself is independent of the aging algorithm to be used. Because LRU gives the smaller capacity miss ratio, the address server can better focus on the requests from purged caches and the time for completing all purge operations is shorter than that for FIFO.

5.2 Natural Recovery of Packet Reachability

A Natural recovery (or self-heeling) strategy is to wait simply until invalid entries are expired (over-written) by the aging algorithm. With this strategy, the address cache keep steady operation with full entries and maintain a relatively low cache miss ratio. However, the cache is not fully trustworthy since the invalid entries remain until they are written over by new address information. For FIFO, each cache entry expires in turn and it is independent of the frequency of accesses. Lifetime of the invalid entry to the number of accesses is determined by N_x and it can be described by using Eq. (6). On the other hand, for LRU, entry which is invalid but accessed frequently may never be expired. Thus, FIFO is preferable to LRU.

6. Comparison between LRU/Purge and FIFO/Natural-Recovery

From the above results, the two schemes, purge operation scheme with LRU and natural recovery scheme with FIFO, are preferred. This section shows the comparison between these two schemes with respect to fast recovery of packet reachability. In addition, we reveal applicable domains of them [8].

Figure 6 shows our system configuration model.

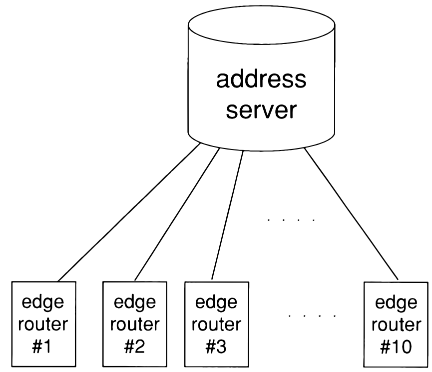


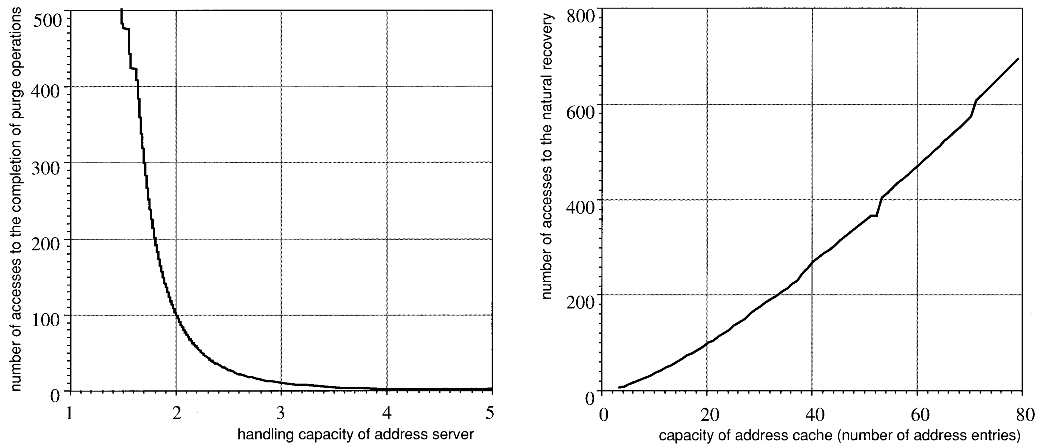
Fig. 6 System configuration.

There are 10 edge routers and an address server. Each edge router has a address cache. All routers are assumed to have the same access rate. The address server provides the following functions:

- It has comprehensive address information.
- It adds address information to an address cache when the address cache miss occurs.
- It can conduct a purge operation to each address cache.

Figure 7(a) shows the relationship between the handling capacity of the address server and the number of accesses which are occurred during the time interval from the time purge for the first cache begin to the time purge for the last cache finished. The figure is calculated by the compulsory miss ration (9) and the capacity miss ratio (5). Interval between adjacent purge operations is determined by both the compulsory miss ratios of already purged caches and capacity miss ratios of the caches waiting purge so that the load of the address server does not reach its capacity. We define the handling capacity of the address server as the number of queries which can be process by the server in unit time. The capacity is 1 if the server can handle all queries from just one address cache. The number of accesses means the number of accesses on one edge router. From Fig. 7(a), the purge scheme with LRU requires long time to complete when the capacity of the address server is low.

Figure 7(b) shows the relationship between the capacity of the address cache and the number of accesses which are generated until completion of the natural recovery (i.e., all cache entries are replaced). This is simply determined by N_x in Eq. (6), since N_x is the number of accesses in the lifetime of an FIFO entry [4] The number of accesses means the number of accesses on one edge router. The figure shows the natural recovery scheme with FIFO requires long time to recover cache validity when the capacity of address caches is large. When the handling capacity of the address server is low, the natural recovery scheme with FIFO is preferable.



(a) (b)

Fig. 7 The number of accesses to recover reachability of packets. (a) Relationship between handling capability of the address server and the number of accesses to the completion of purge operations. (b) Relationship between the capacity of the address cache and the number of accesses to the natural recovery.

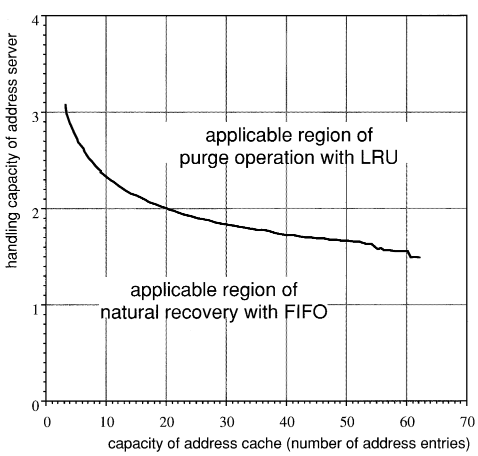


Fig. 8 Applicable domains.

By combining the results from Figs. 7(a) and (b), Fig. 8 shows applicable domains for the two schemes. The vertical axis denotes the handling capacity of the address server and the horizontal axis denotes the capacity of the address cache. Applicable domains are determined by that which scheme can recover packet reachability faster. Note that the applicable region for a purge operation with LRU does not spread under the line such that the handling capacity of the address server equals 1, and it is not only for our system configuration but also for any system configuration. Considering the performance of recent high-speed routers, we think the handling capacity 1 of the address server is a very high figure actually. Assuming the capacity of the address server is less than 1 is safe in many cases, where FIFO with natural recovery works better.

7. Conclusions

This paper have shown the evaluation of the compulsory miss ratio for the address cache and its application.

The evaluation is based on the Dual Zipfian Model describing HTTP access trends. The validity of the evaluation is verified by comparison to experimental results.

Using the evaluation of the compulsory miss ratio, we have investigated some schemes restoring cache validity and their applicable regions. These schemes are a purge operation and natural recovery with aging algorithms.

Note that the applicable region for purge operation spreads only where the handling capacity of the address server is sufficiently high. Some of recent high-speed routers achieves packet throughput of 1 Mpps (packets per second) or more. Address resolution by the address server using a large database of comprehensive address information probably cannot keep up with the packet throughput. The handling capacity of the address server to be over 1 is, therefore, unrealistic in a high-speed network using those high-speed routers.

The natural recovery scheme with FIFO is very simpler than the other three schemes because FIFO is simpler than LRU and the natural recovery strategy is, of course, the simplest strategy. A high performance address server is not required. It is a very interesting point that the natural recovery scheme with FIFO is expected to work most efficiently among the other schemes and is probably a solution in actual network design.

References

[1] M. Laubach and J. Halpern, "Classical IP and ARP over ATM," RFC 2225, 1998.

[2] J.V. Luciani, D. Katz, D. Piscitello, B. Cole, and N. Doraswamy, "NBMA next hop resolution protocol (NHRP)," RFC 2332, 1998.

[3] R. Jain, "Characteristics of destination address locality in computer networks: A comparison of caching schemes," DEC Tech. Rep., DEC-TR-592, 1990.

[4] M. Aida, N. Takahashi, and T. Abe, "A proposal of dual Zipfian model describing HTTP access trends and its application to address cache design," IEICE Trans. Commun., vol.E81-B, no.7, pp.1475–1485, 1998.

[5] S. Ata, M. Murata, and H. Miyahara, "Application of network traffic analysis for the design of high-speed routers," 1999 IEICE TM Workshop, TMWS99-11, 1999.

[6] T. Abe and M. Aida, "Evaluation of address cache miss rate using dual Zipfian model," 1998 IEICE Society Conf., B-7-20, 1998.

[7] M. Aida and N. Takahashi, "Estimation of compulsory miss ratio for address cache using dual Zipfian model," 1998 IEICE Society Conf., B-7-17, 1998.

[8] M. Matsuda and M. Aida, "Replacement policies of address cache for restoring packet reachability," 1998 IEICE Society Conf., B-7-18, 1998.

Appendix A: Analysis of the Access Logs of the WWW Servers and Extension of DZM

A.1 WWW Server-to-User Traffic

In this subsection, we analyze the access logs of the "NTT DIRECTORY" WWW servers† for October, 1997. Those servers accepted in total over a million accesses per day at the time. Through the analysis, we see the characteristics of the server-to-user traffic.

The logs we used here are not the direct records about server-to-user traffic, but are the records of user's requests for the web data on the servers. By using those records of user-to-server requests, we can evaluate the server-to-user traffic with the reasonable assumptions as follow:

- For each request from a user host, the WWW server returns response to the host in some way (mostly requested data, sometimes an error information such as "We don't have that data"). So, we can count the source address of the request as the destination of the server-to-user traffic. The temporal order of the response is mostly same as that of the request.
- One request could trigger multiple responses; text data and graphic data, for example, are often transferred separately by each opening a TCP connection. With regard to analysis of the cache behavior, since occurrence of those multiple responses are temporally close in most cases, we can count those multiple responses as one series which produce single address resolution (cache look-up) process.

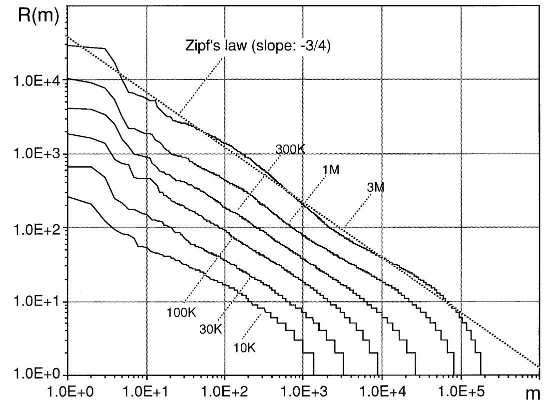


Fig. A·1 Relation between m and $R(m)$.

A.2 Relation between the Frequency of Destination Addresses and the Number of Traffic Exchanges

Let us consider the ordering m of the destination IP addresses of the transfers originated from the WWW server, in descending order of the number of addresses; i.e., we assign the most frequent destination address to $m = 1$, the second most frequent address to $m = 2$, and so on. Then, let us define $R(m)$ the number of transfers sent out to the m-th most frequent address. Figure A·1 represents the distribution of the $R(m)$ for m in a logarithmic scale for the total traffic count of 10,000(10K), 30,000, 100,000, 300,000, 1,000,000(1M), and 3,000,000. The broken straight line indicates the Zipf's law with the slope $-3/4$. The distribution of $R(m)$ shows the good similarity regardless of the total traffic count, and the Zipf's law can be used to describe the distribution well.

A.3 Relation between the Number of Destination Addresses and the Number of the Traffic Exchanges

Let us define $H(n)$ the number of distinct addresses where each address get exactly n transfers. Figure A·2 represents the distribution of the $H(n)$ for n in a logarithmic scale for the total traffic count of 10,000(10K), 30,000, 100,000, 300,000, 1,000,000(1M), and 3,000,000. The distribution of $H(n)$ shows the similarity among the various case of the total traffic count. However, those curved lines of the distribution can not be paralleled by the straight line given by the conventional usage of Zipf's law. This means we can not directly apply the Zipf's law here for the server-to-user traffic, in contrast to the user-to-server traffic which allows direct application of the Zipf's law [4].

† `http://navi.ocn.ne.jp/`

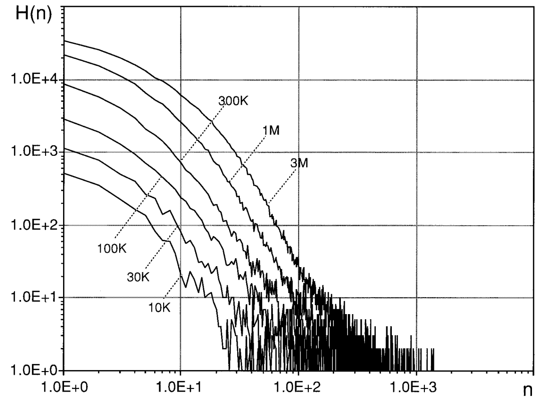


Fig. A·2 Relation between n and $H(n)$.

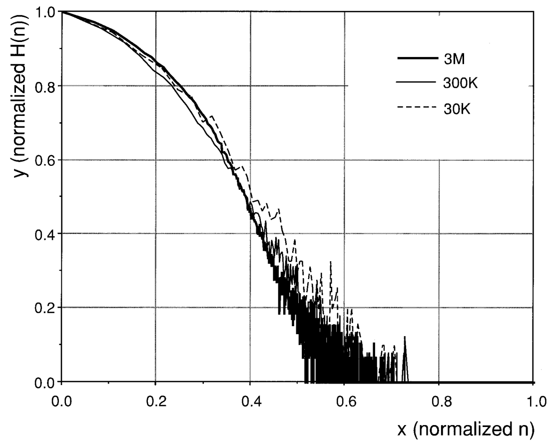


Fig. A·3 Normalized relation between n and $H(n)$.

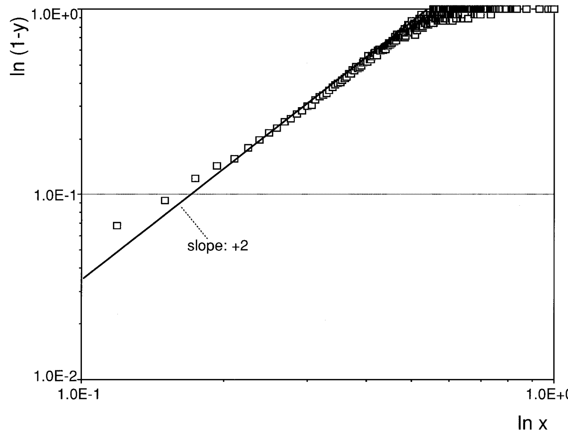


Fig. A·4 Upper-tail behavior of $H(n)$.

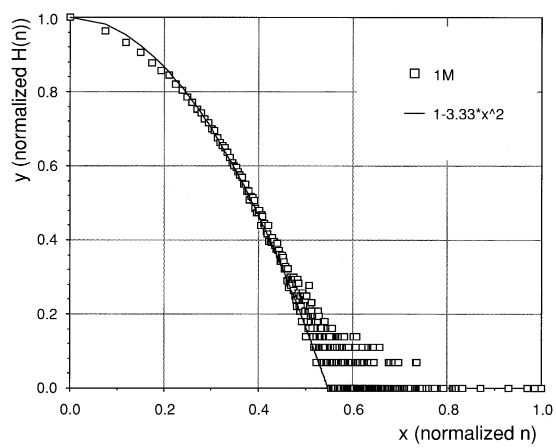


Fig. A·5 Approximation of the upper-tail behavior of $H(n)$.

A.4 Behavior of the Upper-Tail of $H(n)$

The distribution of $H(n)$ for n, where $1 \leq n \leq R(1)$, is $1 \leq H(n) \leq H(1)$. First we introduce the following normalized form of $H(n)$ and $R(n)$:

$$x := \frac{\ln n}{\ln R(1)}, \;\; y := \frac{\ln H(n)}{\ln H(1)} \tag{A·1}$$

such that x and y remain in $0 \leq x, y \leq 1$ regardless of n. Figure A·3 shows the relation between x and y for various n of 30K, 300K, and 3M. Distributions for the different n show a general similarity for the upper-tail (where n is less than about 0.4; the left half in Fig. A·3) of the distributions.

To examine the upper-tail behavior, Fig. A·4 gives the relationship between x (horizontal axis) and $1-y$ (vertical axis), in a logarithmic scale. The relation is well fit by the straight-line of slope 2, thus the following relation stands:

$$\frac{\ln H(n)}{\ln H(1)} - 1 = -a \left(\frac{\ln n}{\ln R(1)}\right)^2 \tag{A·2}$$

where a is a constant. Accordingly, the upper-tail behavior of $H(n)$ can be described as follows:

$$H(n) = H(1)^{1-a\left(\frac{\ln n}{\ln R(1)}\right)^2} \tag{A·3}$$

Figure A·5 demonstrates the approximation described in this section, which has a form of $y = 1 - a\,x^2$, to the actual behavior of $H(n)$ (for the total traffic count of 1M). The value of the parameter a is:

$$a \simeq 10/3 \tag{A·4}$$

The upper-tail behavior of $H(n)$ is approximated well.

A.5 Extension of Dual Zipfian Model

Let N be the total count of the traffic from the server for the given time period, and M be the number of distinct destinations. By utilizing (A·3), the Dual Zipfian Model, Eqs. (1)–(3), is expressed extendedly as follows:

$$M = H \sum_{n=1}^{\lfloor R(1)^{1/\beta} \rfloor} \frac{1}{H^{a\left(\frac{\ln n}{\ln R}\right)^2}} + R^{(\beta-1)/(\beta\, a_r)} \tag{A·5}$$

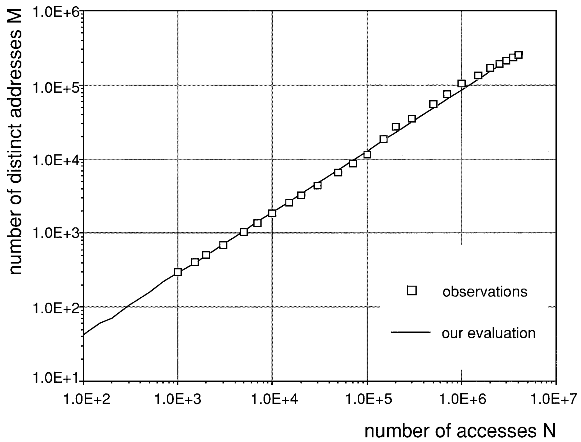


Fig. A·6 Relationship between the traffic count and the number of destinations.

$$N = H \sum_{n=1}^{\lfloor R(1)^{1/\beta} \rfloor} \frac{n}{H^{a\left(\frac{\ln n}{\ln R}\right)^2}} + R \sum_{m=1}^{\lfloor R^{(\beta-1)/(\beta a_r)} \rfloor} \frac{1}{m^{a_r}} \qquad (A\cdot 6)$$

$$N = R \sum_{m=1}^{\lfloor M \rfloor} \frac{1}{m^{a_r}} \qquad (A\cdot 7)$$

where

$$a \simeq 10/3, \;\; a_r \simeq 3/4 \qquad (A\cdot 8)$$

stand and β is a parameter, and $\lfloor\cdot\rfloor$ denotes a floor (integrize) operation.

The relationship between N and M are described well by the extended Dual Zipfian Model (A·5)–(A·7), as shown in Fig. A·6. The horizontal axis is for N in a logarithmic scale, and the vertical one for M also in a logarithmic scale. β in this case is 2.30. Here stands

$$M = N^{5/6} \qquad (A\cdot 9)$$

in the rough.

Masaki Aida received his B.S. and M.S. degrees in Theoretical Physics from St. Paul's University, Tokyo, Japan, in 1987 and 1989, respectively, and received the Ph.D. degree in Telecommunication Engineering from the University of Tokyo, Japan, in 1999. Since he joined NTT Laboratories in 1989, he had been mainly engaged in research on traffic issues in ATM networks and computer communication networks until March 1998. He is currently a manager at Traffic Research Center, NTT Advanced Technology Corporation (NTT-AT). His current interests include traffic issues in multimedia and mobile communication systems. He received the Young Engineer Award of IEICE in 1996. Dr. Aida is a member of Operations Research Society of Japan.

Noriyuki Takahashi received his B.S. and M.S. degrees in Information Science from Kyoto University, Japan, in 1990 and 1992, respectively. He joined NTT LSI Laboratories in 1992, where he has been conducting research on computer-aided design/testing for LSI. Since he joined NTT Information Sharing Platform Laboratories in 1996, he had been mainly engaged in research on routing issues in IP-type networks. He is currently a research engineer at NTT Network Innovation Laboratories. His current interests include high-speed wide-area networking, multimedia communication. He is a member of IPSJ.

Michiyo Matsuda received her B.A. degree in Modern History of Japan from Kokugakuin University, Tokyo, Japan, in 1992. Since she joined NTT Advanced Technology Corporation (NTT-AT) in 1992, she had been mainly engaged in research and development on communications software and distributed caching systems. She is currently an assistant manager at Traffic Research Center, NTT-AT.