

Stochastic Model of Internet Access Patterns*

Masaki AIDA[†] and Tetsuya ABE^{††}, *Regular Members*

SUMMARY This paper investigates the stochastic property of the packet destinations and proposes an address generation algorithm which is applicable for describing various Internet access patterns. We assume that a stochastic process of Internet access satisfies the stationary condition and derive the fundamental structure of the address generation algorithm. Pseudo IP-address sequence generated from our algorithm gives dependable cache performance and reproduces the results obtained from trace-driven simulation. The proposed algorithm is applicable not only to the destination IP address but also to the destination URLs of packets, and is useful for simulation studies of Internet performance, Web caching, DNS, and so on.

key words: *Internet, stochastic process, destination address, LRU stack, caching*

1. Introduction

The Internet system is made of many queueing and caching mechanisms. They are hierarchically configured in protocol layers or their time-scales and are mutually related. In addition, these complicated situations influence the quality of service (QoS) and the performance of systems. In this situation, computer simulations and experimental test-beds probably are powerful approaches to evaluate QoS and Internet performance.

These approaches require a pseudo traffic generator which is based on a traffic model of Internet access behavior. Interarrival distribution of the packets and packet length distribution are a significant part of the traffic model. They have been investigated in [1], [2]. The other significant part of the traffic model is the assignment of the appropriate destination address to a packet generated at a pseudo traffic generator. This is because the cache performance is strongly influenced from the diversity of the destination addresses of the packets.

The fact that computer memory references exhibit locality behavior is now well established [3]. Raj Jain reported that the frames on computer networks also exhibit the locality behavior [4]. Since the cache performance is influenced by the locality behavior, the locality behaviors of the packets are important for eval-

uating Internet performance. In general, the comparison among performances of the different caching mechanisms are conducted through trace-driven simulations [4]–[6]. Trace-driven simulation uses logs of actual Internet accesses as an input traffic of the simulation. Hence, we can reproduce the situation of the specific network traffic by using the trace-driven simulation, but cannot obtain information about the other types of networks.

This paper investigates stochastic property of the destinations of packet and proposes an address generation algorithm which is applicable for describing various Internet access patterns. We assume a stochastic process of Internet access satisfies a sort of the stationary condition and derive the fundamental structure of the address generation algorithm. Pseudo IP-address sequence generated from our algorithm gives dependable cache performance and reproduces the results obtained from trace-driven simulation. The proposed algorithm is applicable not only to the destination IP address but also to the destination URLs of packets, and is useful for simulation studies of Internet performance, Web caching, DNS, and so on.

2. Preliminary

2.1 Notations and Definitions

Internet access behaviors include similar issues in computer memory access patterns. We, therefore, prepare the framework for Internet access models, using well-known notions in computer memory access models.

Time t is incremented to $t + 1$ when an access occurs, and $t = 0, \pm 1, \pm 2, \dots$

- working set: $W(t, \tau)$
The set whose elements are distinct addresses generated during a period $[t - \tau, t)$ (i.e., $\{t - \tau, t - \tau + 1, \dots, t - 1\}$). In case for $\tau < 0$, the period is $(t, t - \tau]$.
- working set size: $w(t, \tau)$
Size of a working set $W(t, \tau)$, i.e., the number of elements of $W(t, \tau)$.
- inverse stack growth function (ISGF) [3]: $f(t)$
Expectation value of the number of distinct addresses generated during a period $(t, t + \tau]$,

$$f(t, \tau) := E[w(t, -\tau)]. \quad (1)$$

Manuscript received January 23, 2001.

Manuscript revised March 15, 2001.

[†]The author is with NTT Information Sharing Platform Laboratories, Musashino-shi, 180-8585 Japan.

^{††}The author is with NTT EAST R&D Center, Tokyo, 163-8019 Japan.

*A part of this paper was presented at IEEE INFOCOM 2001 [19].

In addition, stack growth function (SGF), $g(t, k)$, denotes the expectation value of the number of accesses such that the number of distinct addresses is k , i.e.,

$$f(t, \tau) = k \quad \leftrightarrow \quad g(t, k) = \tau, \quad \text{or} \quad (2)$$

$$g = f^{-1}. \quad (3)$$

Note that because ISGF f is defined only on integer, SGF g cannot be defined directly. The consistent way to obtain the relation between ISGF and SGF is shown in [3]. Hereafter, we regard ISGF and SGF as functions defined on the real number with respect to τ and k , respectively.

2.2 Reference Models for Computer Memory

In computer memory references, it is well known that the concept about *locality* of a reference pattern appears. The locality (especially, temporal locality) implies a high probability of reuse. A similar concept also appears in Internet access [4].

This subsection reviews typical memory reference models arranged for Internet access models [4].

- independent reference model (IRM)
This model assumes accesses are independent. The probability that a new access has address i is determined by the address i (and the probability is denoted as p_i). Accessed address sequence is generated by i.i.d. and the distribution p_i . Because recently accessed addresses do not give any information about the next accessed address, IRM cannot capture locality.
- working set model (WS model) [8]
This model assumes that the addresses accessed in the last ϖ accesses are highly likely to be accessed. The WS model, therefore, captures locality. The interval ϖ is called the working set window size. For a given working set window size ϖ , the smaller working set size $w(t, \varpi)$ means stronger locality.
- least recently used (LRU) stack model [9]
An LRU stack is a list of addresses sorted in order according to the times of their most recent access. So, the most recently accessed address is at the top of the stack and the least recently accessed address is at the bottom. The probability that the newly accessed address is the same as the address at k -th position in the LRU stack is determined by position k , and its probability is denoted as a_k . Usually, a_k decreases with respect to increases in k . Because the most recently accessed address is at the top of the LRU stack, the probability that the next accessed address is the same is high. The LRU stack model, therefore, captures locality.

In performance evaluation of address resolutions,

the performance of the address cache is an essential object [4], [7]. Cache performance strongly depends on locality of Internet accesses. Because IRM cannot take locality into account, it is an insufficient model for Internet access modeling.

Since we can evaluate cache performance if ISGF, f , is given [7], ISGF and WS models are important. Hereafter, from natural and basic assumptions on ISGF and investigation of the working set, we show that the address generation probability must obey an LRU stack model. In addition, we show the relationship between the address generation probability a_k and ISGF f .

3. Basic Structure of Address Generation Probability

3.1 Assumption

We assume the following property of ISGF.

Time-translation invariance:

ISGF $f(t, \tau)$ is independent of the time t which denotes time when we start to measure address generation. For any t, s ,

$$f(t, \tau) = f(s, \tau). \quad (4)$$

From this assumption, we denote $f(\tau) := f(t, \tau)$.

Validation of time-translation invariance is discussed in Sect. 5 through experimental data.

In queueing models, the typical and usual traffic model as an input is described by a stationary stochastic process. On the other hand, the notion of a *stationary process* of address generation is not well-defined yet. However, it is natural to accept that if the address generation process is *stationary*, the ISGF should have a time-translation invariance. Thus, the time-translation invariance should be, at least, a necessary condition of the stationarity. We assume it in order to produce a stationary process of the address generation.

Incidentally, we can consider other possibilities for the stationarity condition. For example, we may adopt the assumption that the working set size distribution [10],

$$p(k, \tau) := \Pr\{w(t, \tau) = k\}, \quad (5)$$

is independent of t . Although this seems to be a stronger stationarity condition than the time-translation invariance of ISGF f , stationarity of (5) is obtained from (4) as a result. This is because the assumption (4) completely determines the fundamental structure of the address generation algorithm, as shown in the following sections.

3.2 Time-Reversal Invariance

Let us consider $f(-\tau)$. This means the expectation

value of the number of distinct addresses generated in the period which is from a time t until the past time $t - \tau$ through the backward direction of time. By using the time-translation invariance (4), we have the time-reversal invariance for ISGF,

$$\begin{aligned} f(-\tau) &= f(t, -\tau) \\ &= f(t - \tau - 1, \tau) \\ &= f(\tau). \end{aligned} \tag{6}$$

3.3 Address Generation Probability

We consider three adjacent periods A , B , and C (Fig. 1). Let the sets of distinct addresses generated in these periods, i.e., working set, be $W(A)$, $W(B)$, and $W(C)$, respectively. Figure 2 denotes Venn's diagram of these working sets.

Here, we focus on the subset W^* whose elements are also elements of both $W(A)$ and $W(C)$, but are not elements of $W(B)$, i.e., the hatched part in Fig. 2,

$$W^* := \{W(A) \cap W(C)\} \setminus W(B). \tag{7}$$

The size of W^* is obtained as

$$\begin{aligned} |W^*| &= |\{W(A) \cap W(C)\} \setminus W(B)| \\ &= \{|W(B) \cup W(C)| - |W(B)|\} \\ &\quad - \{|W(A) \cup W(B) \cup W(C)| \\ &\quad - |W(A) \cup W(B)|\}. \end{aligned} \tag{8}$$

We can choose m , n and 1 accesses as the periods A , B , and C , respectively (Fig. 3). Then we have

$$\begin{aligned} |W^*| &= \{w(t, n + 1) - w(t - 1, n)\} \\ &\quad - \{w(t, n + m + 1) - w(t - 1, n + m)\}. \end{aligned} \tag{9}$$

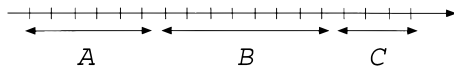


Fig. 1 Time intervals A , B and C .

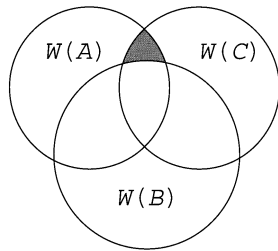


Fig. 2 Venn's diagram of working sets.

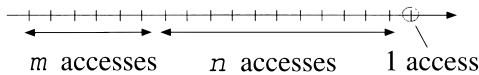


Fig. 3 Time intervals.

Applying the time-reversal invariance, we have

$$\begin{aligned} E[|W^*|] &= \{f(-(n + 1)) - f(-n)\} \\ &\quad - \{f(-(n + m + 1)) - f(-(n + m))\} \\ &= \{f(n + 1) - f(n)\} \\ &\quad - \{f(n + m + 1) - f(n + m)\}. \end{aligned} \tag{10}$$

The physical meanings of $E[|W^*|]$ are as follows. $|W^*|$ is 1 only when the address X generated at a time (period C) is not in $W(B)$ but in $W(A)$. Otherwise, $|W^*|$ is 0. Thus, $|W^*|$ is a random variable and can be denoted using indicator function $\mathbf{1}\{\cdot\}$ by

$$|W^*| = \mathbf{1}\{X \notin W(B), X \in W(A)\}. \tag{11}$$

Therefore, $E[|W^*|]$ means the probability of $|W^*| = 1$,

$$E[|W^*|] = \Pr\{X \notin W(B), X \in W(A)\}. \tag{12}$$

Next, we choose n such as $f(n) = k - 1$ ($k = 1, 2, \dots$) and choose m such as $f(n + m) = k$. Substituting $n = g(k - 1)$ and $n + m = g(k)$, (10) gives

$$\begin{aligned} E[|W^*|] &= \{f(g(k - 1) + 1) - (k - 1)\} \\ &\quad - \{f(g(k) + 1) - k\}. \end{aligned} \tag{13}$$

Equation (13) means the probability that the newly accessed address, X , is identical with the k -th most recently accessed address.

Address generation probability:

Here, we define a_k as

$$\begin{aligned} a_k &:= \{f(g(k - 1) + 1) - (k - 1)\} \\ &\quad - \{f(g(k) + 1) - k\}. \end{aligned} \tag{14}$$

This means that the probability that X is identical with the most recently accessed address is a_1 , the probability that X is identical with the 2nd most recently accessed address is a_2 , ..., and the probability that X is identical with the k -th most recently accessed address is a_k . In other words, address generation must obey the LRU stack model whose probability is (14).

If $\{a_k; k = 1, 2, \dots\}$ is the probability, ISGF f must satisfy $a_k \geq 0$ and

$$\sum_{k=1}^{\infty} a_k = 1, \tag{15}$$

that is, $\{f(g(k) + 1) - k\}$ is monotonously decreasing with respect to increases in k ($= 0, 1, 2, \dots$) and satisfies

$$\lim_{k \rightarrow \infty} \{f(g(k) + 1) - k\} = 0. \tag{16}$$

4. Address Generation Algorithm

This section shows an address generation algorithm by applying the results of the above section. The proposed algorithm gives a pseudo-address sequence, a sequence of integer.

4.1 Address Generation Procedure

We define the generative LRU stack vector $\mathbf{L}(i, m)$ as follows. At the time immediately before i -th ($i = 0, 1, 2, \dots$) address generation, we denote the most recently accessed address as x_1 , the 2nd most recently accessed address as x_2, \dots , and the k -th most recently accessed address as x_k . Then

$$\mathbf{L}(i, m) := \{x_1, x_2, \dots, x_m\}. \quad (17)$$

Here, the number of components of $\mathbf{L}(i, m)$, m , is called the depth of $\mathbf{L}(i, m)$. Initially, the depth is chosen as 0, i.e., $m = 0$ for $\mathbf{L}(0, 0)$. Then m means the number of distinct address generations.

Let X_i denote the address of the i -th access. Address sequence $\{X_i; i = 0, 1, 2, \dots\}$ is generated by the following procedure.

1. Initially, we choose $X_0 = 1$, $\mathbf{L}(1, 1) = \{1\}$ and $i = 1$.
2. Determine the number j as a realization of the i.i.d. random variable J which obeys the distribution

$$\Pr\{J = k\} = a_k \quad (k = 1, 2, 3, \dots), \quad (18)$$

by using (14).

3. For the depth m of $\mathbf{L}(i, m)$, if $m < j$, then assign the address of new access as

$$X_i = m + 1, \quad (19)$$

and update $\mathbf{L}(i+1, m+1)$ and increment $i \leftarrow i+1$. Return to 2.

4. For the depth m of $\mathbf{L}(i, m)$, if $m \geq j$, then assign the address of new access as

$$X_i = x_j, \quad (20)$$

and update $\mathbf{L}(i+1, m)$ and increment $i \leftarrow i+1$. Return to 2.

Because the number j can be obtained using a binary search algorithm, high-speed address generation is possible.

4.2 Reproduction of the ISGF Behavior

When the depth of the generative LRU stack vector $\mathbf{L}(t, m)$ is m at a time t , we consider the probability that the working set size $w(t, n)$ increases, i.e.,

$$b_m := \Pr\{w(t+1, n+1) = m+1 | w(t, n) = m\}. \quad (21)$$

This means the generation probability of a brand-new address which does not exist in $\mathbf{L}(t, m)$. So, it is obtained as

$$b_m = \sum_{k=m+1}^{\infty} a_k$$

$$= f(g(m) + 1) - m. \quad (22)$$

Equation (22) is valid for all m ($m = 0, 1, 2, \dots$). The probability of a brand-new address generation depends only on the depth of the generative LRU stack vector. Because the same depth gives the same probability, b_m is independent of the time when we start to measure address generations. Thus, the address sequence obtained from our algorithm reproduces the time-translation invariance of ISGF.

Let us consider the time that the depth m of the generative LRU stack vector $\mathbf{L}(t, m)$ is incremented to $m+1$. Because the time obeys a geometrical distribution, the mean time is $1/b_m$. Therefore, SGF g is obtained as

$$g(k) = \sum_{m=0}^{k-1} \frac{1}{b_m}. \quad (23)$$

We define the slope of f as the difference

$$\left. \frac{\Delta f}{\Delta g} \right|_m := \frac{f(g(m) + 1) - f(g(m))}{(g(m) + 1) - g(m)} = b_m. \quad (24)$$

From (23), we have

$$g(k) = \sum_{m=0}^{k-1} \left[\left. \frac{\Delta f}{\Delta g} \right|_m \right]^{-1}. \quad (25)$$

This means the address sequence obtained from the proposed algorithm reproduces ISGF, f .

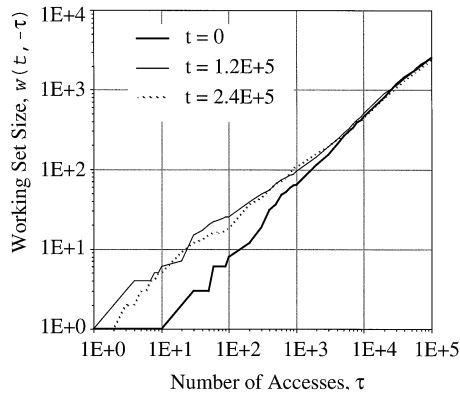
5. Working Set Size Behaviors of Internet Accesses

In order to obtain pseudo-address sequence by using the proposed algorithm in practice, it is necessary to determine ISGF f . In this section, we investigate asymptotic property of working set size behaviors of Internet accesses from experimental data, and show that a power law is widely applicable. Simultaneously, validation of the time-translation invariance is discussed.

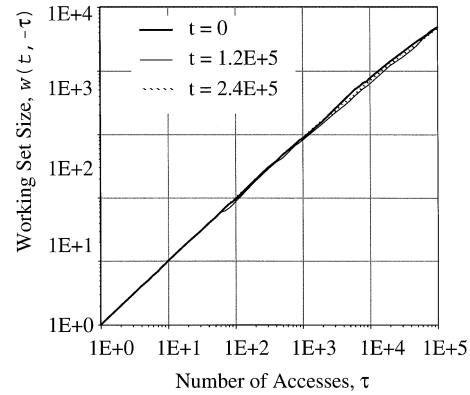
In most cases, it is well known that the destination addresses or URLs are characterized by Zipf-type distributions [7], [5]–[11] (strictly, parts of them are Lotoka-type distributions [12]). Here, an example of the Zipf-type distribution is appears in the later section (see Sect. 7 and Fig. 8(a)). Although, in some cases, we can calculate ISGF by using Zipf-type distribution [7], [13], it is not always feasible. Therefore, it is required to determine ISGF from generally applicable property with respect to Internet accesses.

5.1 Working Set Size Behaviors and Time-Translation Invariance of ISGF

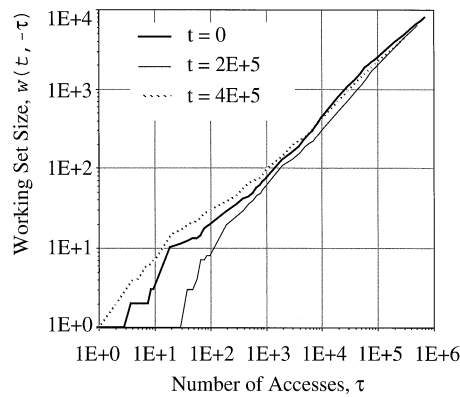
As an example of access networks of the Internet,



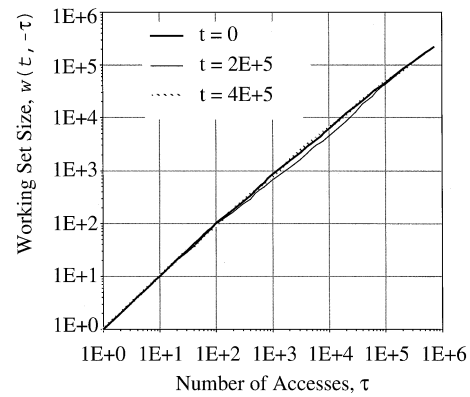
(a) For the destination IP addresses (NTT labs. (Atsugi), Nov. 1996).



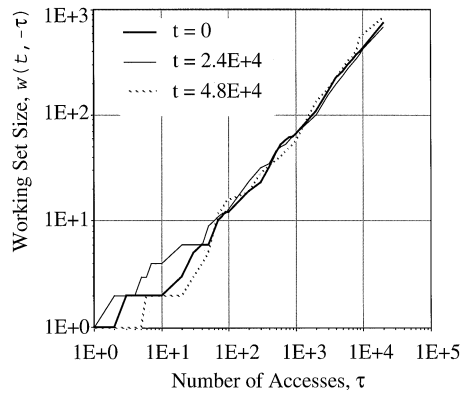
(b) For the destination URLs (NTT labs. (Atsugi), Nov. 1996).



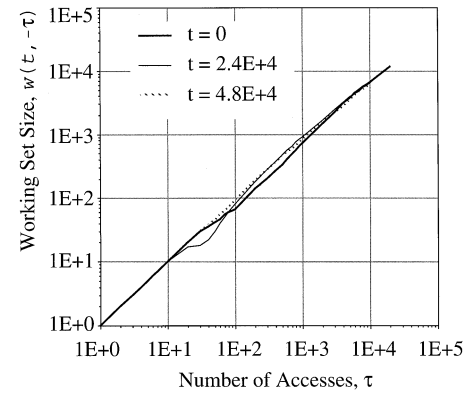
(c) For the destination IP addresses (NTT labs. (Atsugi), March 1997).



(d) For the destination URLs (NTT labs. (Atsugi), March 1997).



(e) For the destination IP addresses (NTT labs. (Yokosuka), March 1997).

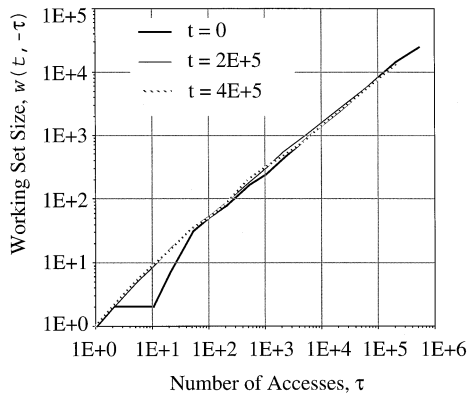


(f) For the destination URLs (NTT labs. (Yokosuka), March 1997).

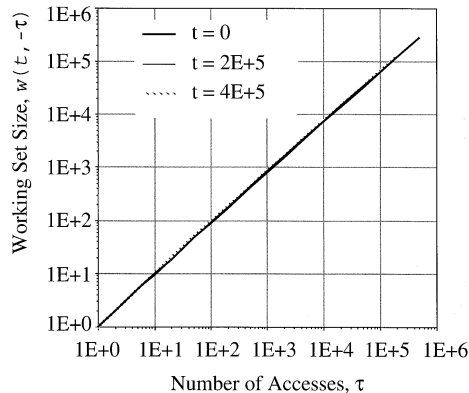
Fig. 4 Working set size behaviors of destinations (NTT labs.).

we investigate access logs of a proxy server at NTT Laboratories. Logs are for 15 successive days. Figure 4(a) shows the relationship between working set size, $w(t, -\tau)$, and the number of accesses, τ , for the destination IP addresses in log-log scale, where logs were obtained from NTT Labs. (at Atsugi, Japan, in Nov. 1996). Each line indicates $w(t, -\tau)$ for $t = 0$, 1.2×10^5 , and 2.4×10^5 , respectively. Similarly,

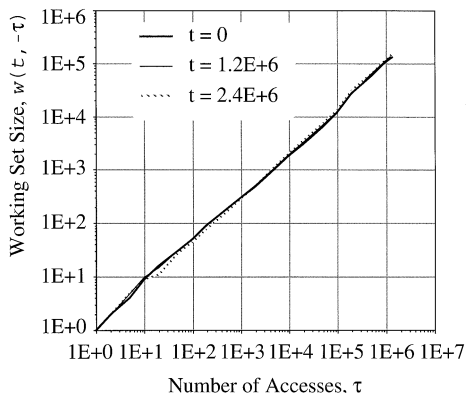
Fig. 4(b) shows the relationship between working set size, $w(t, -\tau)$, and the number of accesses, τ , for the destination URLs in log-log scale, using the same logs. Figures 4(c)–(f) show the similar relationships, where logs were obtained from NTT Labs. (at Atsugi, in March 1997) and NTT Labs. (at Yokosuka, Japan, in March 1997), respectively. Each figure shows $w(t, -\tau)$ is independent of t for large τ . From the law of large



(a) For the destination IP addresses (NLNR, July 30, 1999).



(b) For the destination URLs (NLNR, July 30, 1999).

Fig. 5 Working set size behaviors of destinations (NLNR).**Fig. 6** Working set size behaviors of originations (NTT DIRECTORY, Oct. 1997).

number, asymptotic behavior of the working set size can be related to the behavior of ISGF, as

$$w(t, -\tau) \sim f(t, \tau) \quad (\tau \gg 1). \quad (26)$$

Thus, these figures imply the time-translation invariance of ISGF, at least in the asymptotic region, $\tau \gg 1$.

As an example of backbone networks of the Internet, we investigate access logs of a cache server at NLNR [14]. Logs are for July 30, 1999. Figure 5(a) shows the relationship between working set size, $w(t, -\tau)$, and the number of accesses, τ , for the destination IP addresses in log-log scale. Each line indicates $w(t, -\tau)$ for $t = 0, 2 \times 10^5$, and 4×10^5 , respectively. Similarly, Fig. 5(b) shows the relationship between working set size, $w(t, -\tau)$, and the number of accesses, τ , for the destination URLs in log-log scale. These figures also imply the time-translation invariance of ISGF, at least in the asymptotic region, $\tau \gg 1$.

As an example of WWW servers of the Internet, we investigate access logs of a directory site, NTT DIRECTORY [15]. Logs are for 31 successive days in October 1997. Figure 6 shows the relationship between working set size, $w(t, -\tau)$, and the number of accesses, τ , for

the origination IP addresses in log-log scale. Each line indicates $w(t, -\tau)$ for $t = 0, 1.2 \times 10^6$, and 2.4×10^6 , respectively. This figure also implies the time-translation invariance of ISGF, at least in the asymptotic region, $\tau \gg 1$.

5.2 Asymptotic Power Law

From the above graphical observations, we can recognize that $f(\tau)$ satisfies a power law with respect to τ for $\tau \gg 1$,

$$f(\tau) \simeq \tau^\alpha \quad (\tau \gg 1), \quad (27)$$

where α is a constant. A similar property is known in computer memory access [3]. This property is widely applicable, e.g., for access networks, backbone networks, at a WWW server, and for the destination IP addresses and URLs.

6. Address Generation Algorithm Using Power Law

This section investigates pseudo-address generation by applying the power law of ISGF.

6.1 Address Generation Probability

We assume the asymptotic power law for $f(n)$ is applicable for all n , i.e.,

$$f(n) = n^\alpha. \quad (28)$$

Then, address generation probability (14) is obtained as

$$a_k = \left\{ \left((k-1)^{1/\alpha} + 1 \right)^\alpha - (k-1) \right\} - \left\{ \left(k^{1/\alpha} + 1 \right)^\alpha - k \right\}. \quad (29)$$

Because $a_k > 0$ and

$$\begin{aligned} \sum_{k=1}^{\infty} a_k &= 1 - \lim_{k \rightarrow \infty} \left\{ (k^{1/\alpha} + 1)^\alpha - k \right\} \\ &= 1 \end{aligned} \quad (30)$$

for $0 < \alpha < 1$, $\{a_k; k = 1, 2, \dots\}$ is probability. The asymptotic behavior of a_k is expressed as

$$a_k = (1 - \alpha) k^{-1/\alpha} \quad (k \gg 1). \quad (31)$$

6.2 Time-Translation Invariance and Asymptotic Power Law

Using power law (28), the probability that the working set size increases, (22), is expressed as

$$\begin{aligned} b_m &= \sum_{k=m+1}^{\infty} a_k \\ &= \left\{ (m^{1/\alpha} + 1)^\alpha - m \right\}. \end{aligned} \quad (32)$$

Using the asymptotic behavior of b_m ,

$$b_m = \alpha m^{-(1/\alpha-1)} + O(m^{-(2/\alpha-1)}), \quad (33)$$

and (23), we have

$$g(k) = \sum_{m=0}^{k-1} \left\{ \frac{1}{\alpha} m^{1/\alpha-1} + O(m^{-1}) \right\}. \quad (34)$$

Therefore,

$$\lim_{k \rightarrow \infty} \frac{\log g(k)}{\log k} = \frac{1}{\alpha}. \quad (35)$$

This means the address sequence obtained from our algorithm reproduces the asymptotic power law of ISGF,

$$f(n) = n^\alpha \quad (n \gg 1). \quad (36)$$

7. Experimental Results

In order to verify the effectiveness of our algorithm described in Sect. 4, we compare the cache miss ratios. We show the cache miss ratio which is obtained from simulation with address sequences generated by our algorithm as input, and compare it with the results of trace-driven simulation and theoretical values.

As shown in [7], theoretical values of cache miss ratio, for LRU and FIFO, are obtained through the following formulae. For LRU cache, the cache miss ratio, P_L is given by

$$P_L = \frac{1}{g(k+1) - g(k)}, \quad (37)$$

where k denotes the capacity of the cache (the number of address entries). Applying the power law (28), we have

$$P_L = \frac{1}{(k+1)^{1/\alpha} - k^{1/\alpha}}. \quad (38)$$

Incidentally, for $k \gg 1$, (38) is evaluated as

$$P_L \simeq \alpha k^{1-1/\alpha}. \quad (39)$$

This means the behavior of the cache miss ratio obeys a power law, asymptotically. The same behavior is well known in computer memory reference, and called Chow's empirical power law [16], [17]. On the other hand, the cache miss ratio for FIFO is given by

$$P_F = \frac{k}{N_x}, \quad (40)$$

where N_x is the number which satisfies

$$4f(N_x) - 3k - \sqrt{9k^2 - 8k \{f(2N_x) - f(N_x)\}} = 0. \quad (41)$$

The data used in the trace-driven simulation is access logs of a proxy server at NTT Labs. (at Atsugi, in March 1997). We focus on the destination IP addresses in the logs and, in this case, the value of α for the asymptotic power law is 2/3 from graphical observation of Figs. 4(a), (c) and (e).

Figures 7(a) and (b) show the relationship between cache miss ratio and the capacity of the cache, for LRU and FIFO, respectively. The horizontal axis denotes the capacity of the cache and the vertical axis denotes the cache miss ratio. The black plot denotes the results from our algorithm and the white plot denotes the results from trace-driven simulations. The solid line denotes theoretical values. These figures show that our algorithm gives a dependable evaluation.

Next, we compare the distributions of the addresses. Figures 8(a) and (b) show the Zipf-type plots of the destination IP addresses (NTT Labs. at Atsugi, in Nov. 1996) and the pseudo-address sequence obtained from our algorithm, respectively. The horizontal axes denote the rank order of addresses where all addresses are sorted in descending order according to the frequencies of appearances. The vertical axes denote the frequency of the corresponding address appearances (the number of accesses). Figure 8(a) exhibits linearity in the log-log scale. This is called the Zipf-type distribution and means that only some of the addresses appear very frequently. On the other hand, Fig. 8 does not exhibit the concentration of the appearance frequencies. This is because the proposed algorithm is democratic for all addresses in the sense that the probability of an address generation is determined only by the present stack position at the LRU stack. Therefore, the temporal locality of address appearances strongly influence to the cache performances for LRU and FIFO caches, but the shape of the distribution is not.

8. Conclusion and Discussion

This paper has shown the structure of the address

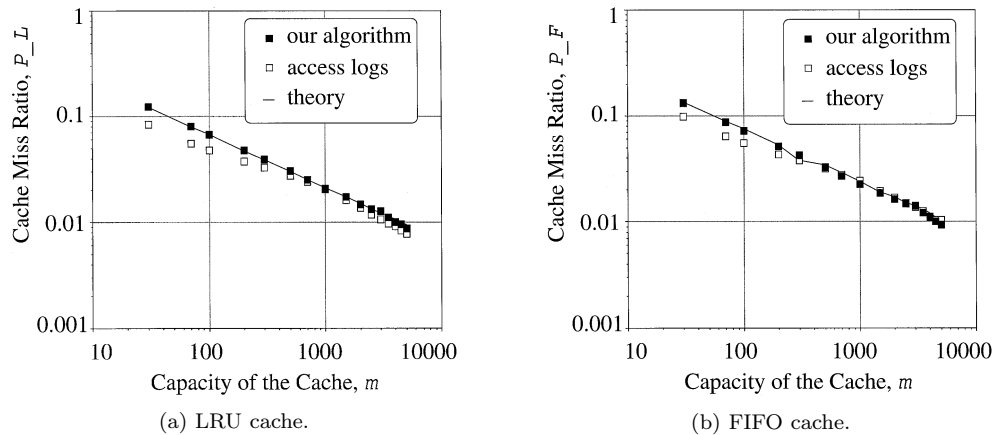


Fig. 7 Relations between the cache miss ratio and the capacity of the cache.

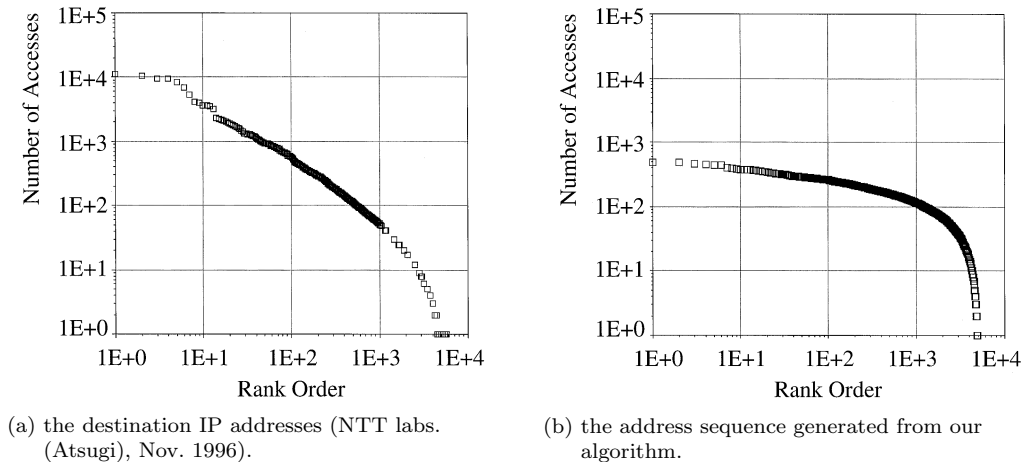


Fig. 8 Zipf-type plots of the destination IP addresses.

generation probability from the assumption of time-translation invariance of ISGF. The structure is an LRU stack model and has a simple relation between address generation probability and ISGF. In particular, we have revealed the following equivalence:

$$\begin{array}{c} \text{address generation process whose ISGF has a} \\ \text{time-translation invariance.} \\ \Updownarrow \\ \text{LRU stack model whose probability is (14).} \end{array}$$

This paper also has shown the general property of ISGF, that is, the asymptotic power law is applicable to ISGF in a wide number of Internet access cases. In addition, by using these results, we have studied properties of the pseudo-address sequence obtained from our algorithm. This address sequence gives dependable evaluations of the cache miss ratio.

The proposed algorithm is democratic for all addresses in the sense that the probability of an address generation is determined only by the present stack position at the LRU stack. In actual Internet usage, some addresses are accessed with extremely high frequen-

cies. This nature is often characterized by Zipf-type distributions. The address sequence generated from the proposed algorithm, however, does not reproduce these distributions. Let us consider an aging algorithm which writes the new address over the least frequently accessed entry. If we choose this aging algorithm for a cache, the address sequence obtained from our algorithm may not give a dependable evaluation. Of course, because this aging algorithm does not take locality into consideration, we can choose IRM for an address generation algorithm.

There are probably two reasons why our approach cannot reproduce Zipf-type distributions. Issues we should study are listed as follows:

- Validation of the time-translation invariance of ISGF.
- Validation of (28) assumed in (29). Equation (28) is only an asymptotic property.

The time-translation invariance seems to be a natural assumption. If we denied the assumption, the structure of the address generations loses the Markovian

property whose state is described by the generative LRU stack vector. This may cause a situation where complicated approaches are required. On the other hand, it is preferable to have the power law adopted for ISGF replaced by a more strict relationship. However, the behavior of ISGF is probably inessential to reproduce Zipf-type distributions. In order that both time-translation invariance and Zipf-type distributions are compatible, an example of extension for the proposed algorithm has been discussed in [18]. These issues are for further study.

References

- [1] C.R. Cunha, A. Bestavros, and M.E. Crovella, "Characteristics of WWW client-based traces," Tech. Rep. BU-CS-95-010, Boston Univ., 1995.
- [2] M. Nabe, K. Baba, M. Murata, and H. Miyahara, "Analysis and modeling of WWW traffic for designing Internet access networks," IEICE Trans., vol.J80-B-I, no.6, pp.428-437, June 1997.
- [3] M. Kobayashi and M.H. MacDougall, "The stack growth function: Cache line reference models," IEEE Trans. Comput., vol.38, no.6, pp.798-805, 1989.
- [4] R. Jain, "Characteristics of destination address locality in computer networks: A comparison of caching schemes," DEC Tech. Rep., DEC-TR-592, 1990.
- [5] S. Glassman, "A caching relay for World Wide Web," Comput. Networks and ISDN Systems, vol.27, pp.165-173, 1994.
- [6] N. Osawa, F. Hayano, T. Yuba, and K. Hakozaiki, "Evaluation of replacement policies in WWW cache based on logs from proxy server," Tech. Rep. IPSJ, DPS74-33, 1996.
- [7] 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, July 1998.
- [8] P.J. Denning and S.C. Schwartz, "Properties of the working-set model," Commun. ACM, vol.15, no.3, pp.191-198, 1972.
- [9] J.E. Shemer and G.A. Shippey, "Statistical analysis of paged and segmented computer systems," IEEE Trans. Electronic Computers, vol.EC-15, no.6, pp.855-863, 1966.
- [10] R. Turner and B. Strecker, "Use of the LRU stack depth distribution for simulation of paging behavior," Commun. ACM, vol.20, no.11, pp.795-798, 1977.
- [11] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," Proc. IEEE INFOCOM'99, March 1999.
- [12] R. Rousseau and S. Rousseau, "Informetric distributions: A tutorial review/Les distributions informétriques: Un aperçu," Canadian J. Information and Library Science, vol.18, no.2, pp.51-63, 1993.
- [13] M. Aida, N. Takahashi, and M. Matsuda, "Evaluation of compulsory miss ratio for address cache and replacement policies for restoring packet reachability," IEICE Trans. Commun., vol.E83-B, no.7, pp.1400-1408, July 2000.
- [14] National laboratory for applied network research (NLNR)/Measurement and operations analysis team (MOAT), <http://moat.nlanr.net>.
- [15] NTT DIRECTORY, <http://navi.ocn.ne.jp>.
- [16] C.K. Chow, "Determination of cache's capacity and its matching storage hierarchy," IEEE Trans. Comput., vol.C-25, no.2, pp.157-164, 1976.
- [17] D. Thiébaud, "On the fractal dimension of computer programs and its application to the prediction of the cache miss ratio," IEEE Trans. Comput., vol.38, no.7, pp.1012-1026, 1989.
- [18] M. Aida and T. Abe, "Stochastic process of packet destination addresses describing Internet access patterns," IEICE Technical Report, IN2000-74 (2000-09), 2000.
- [19] M. Aida and T. Abe, "Pseudo-address generation algorithm of packet destinations for Internet performance simulation," Proc. IEEE INFOCOM 2001, April 2001.



Masaki Aida received his B.Sc. and M.Sc. degrees in Theoretical Physics from St. Paul's University, Tokyo, Japan, in 1987 and 1989, 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.

From March 1998 to March 2001, he was a manager at Traffic Research Center, NTT Advanced Technology Corporation (NTT-AT). He is currently a Senior Research Engineer at NTT Information Sharing Platform Laboratories. His current interests include traffic issues in communication systems. He received the Young Engineer Award of IEICE in 1996. Dr. Aida is a member of the IEEE and the Operations Research Society of Japan.



Tetsuya Abe received his B.Sc. degree in Physics from Keio University, Yokohama, Japan, in 1988. He joined NTT Laboratories in 1988, where he has been conducting research on process-device modeling, development on on-chip RISC controller and designing of high-speed Network-Nodes. Since 2000, he has been an associate manager at NTT EAST R&D center. His current interests include architecture of broadband networks. He

received the Best Industrial ASIC Award of ED&TC in 1996.