Measuring the Size of the Internet via Importance Sampling

Song Xing and Bernd-Peter Paris

***Abstract*—Measuring the size of the Internet via Monte Carlo sampling requires probing a large portion of the Internet protocol (IP) address space to obtain an accurate estimate. However, the distribution of information servers on the Internet is highly nonuniform over the IP address space. This allows us to design probing strategies based on importance sampling for measuring the prevalence of an information service on the Internet that are significantly more effective than strategies relying on Monte Carlo sampling. We present thorough analysis of our strategies together with accurate estimates for the current size of the Internet Protocol Version 4 (IPv4) Internet as measured by the number of publicly accessible web servers and FTP servers.**

***Index Terms*—Importance sampling, Monte Carlo sampling, size of the Internet.**

I. Introduction

As COMPUTERS and communication networks have become faster and more widespread, the Internet has experienced tremendous growth since its inception. Unlike the telephone network which was designed in a centralized way by major corporations, the Internet design emphasizes decentralized control. Though it is essential to the Internet's scalability and robustness, the decentralization of control causes problems that may hamper the evolution of the Internet, including unreliable service or nonoptimal routing.

A more pernicious problem is that it is difficult to determine how large the Internet really is, i.e., to quantify exactly how many hosts are currently on the Internet. Therefore, it is difficult to estimate reliably the growth of the Internet and predict, for example, when the available Internet address will eventually run out. Hence, developing efficient means for assessing the size of the Internet, is of interest, for example, for network engineering or network capacity planning purposes.

There are relatively few publications on measuring the size of the Internet. The Internet Software Consortium, for example, attempts to discover every host on the Internet by querying the domain name system (DNS) [1]. The problem with this approach is that it is inaccurate since a host name with an assigned IP address does not mean the host actually exists. Conversely, a host does not have to be in the DNS to communicate, thus a second "ping" step may be needed to obtain the number of live hosts. This approach is also inefficient as it requires several days to collect data, and it may not be scalable as the Internet continues to grow. In fact, the survey conducted by the Internet Software Consortium may be well suited to take advantage of the methods described herein.

Netcraft does a periodic survey of web server software usage on the Internet and the number of web servers [2]. Their statistics are obtained by collecting and collating the host names providing the HTTP service, systematically polling each one with an HTTP request for the server name, and looking in detail at the network characteristics of the HTTP replies. Obviously, this approach is time-consuming collection of the data and the accuracy of their survey depends on the number of data collected.

In this work, we emphasize our importance-sampling based method over actual measurements. Nevertheless, to demonstrate the usefulness of our approach, we report our measurements of an important part of the current Internet. Specifically, we are measuring the number of hosts connected to the public Internet (hosts with a publicly routable IP address) providing a given information service such as WWW or FTP. As will be explained below, our methods are based on sampling the Internet protocol (IP) address space. Hence, our methods have their own shortcomings, including an inability to distinguish between multiple web domains hosted by the same server (virtual hosting). Similarly, we would not be able to tell that a system of servers employing some form of load balancing should probably be counted as only a single server. Because of these differences, it should be expected that our results are quite different from those obtained by Netcraft [2] for example.

The primary strengths of the methods proposed herein are simplicity, wide applicability, and scalability. The sampling based strategies consist only of an address generator that determines which IP addresses are to be probed, the probing client itself, and a simple analysis system for tallying the results of the probes. Our methods are widely applicable to network applications following the client-server paradigm. For each such application, only the probing client would have to be altered. The results could be used to track the prevalence and growth of a network application or the rate of adoption of a new protocol. Similarly, if probes employ some form of *echo request* the size of the entire public Internet may be measured. Perhaps, most importantly, we believe that our methods are able to keep up with the continued explosive growth of the Internet. Since importance-sampling allows us to focus measurements on the most relevant part of the address space, we anticipate that measurement methods based on importance sampling will scale with the size of the address space.

This paper principally proposes and investigates novel, efficient and effective methods based on importance sampling for

Manuscript received August 18, 2002; revised March 5, 2003.

The authors are with the Department of Electrical and Computer Engineering, George Mason University, Fairfax, VA 22030 USA (e-mail: sxing@gmu.edu; pparis@gmu.edu).

Digital Object Identifier 10.1109/JSAC.2003.814510

0733-8716/03$17.00 © 2003 IEEE

measuring the size of the Internet. Consequently, we first provide some preliminaries on Monte Carlo and importance sampling for measuring the size of the Internet. Next, the optimal unbiased measurement strategy based on importance sampling is introduced. We demonstrate in Section IV that even better strategies are possible if the restriction of *absolutely continuous* biasing strategies is dropped. Measurement results for our importance sampling approaches are presented and compared with Monte Carlo sampling. In Section VI, we describe some of our measurement results for the size and growth of the Internet.

II. Preliminaries

A naive way to accomplish our objective of measuring the prevalence of a given information service on the Internet would be to probe the entire IP address space and count the number N_w of information servers thus found. We could express this procedure mathematically by the equation

$$N_w = \sum_{n=1}^{2^{32}} I(A_n) \tag{1}$$

where the upper limit of the sum reflects the size of the IP address space and $I(A_n)$ indicates the result of probing address A_n. To evaluate $I(A_n)$, a probe is sent to the IP address A_n and if the response to the probe is positive (e.g., indicates the presence of a server at address A_n) $I(A_n)$ assumes the value 1. Otherwise, $I(A_n) = 0$. For measuring the number of World Wide Web (WWW) servers on the public Internet, we would send a HTTP `HEAD` request to address A_n and count a success $(I(A_n) = 1)$ if we receive a message with a response status code of `2XX`. Clearly, it can be argued that any other response code would also indicate the presence of a server at that address, but we restricted ourselves to "Success" codes in this work.

We will find it convenient to formulate our results in terms of the quantity P_w defined as

$$P_w = \frac{N_w}{2^{32}}. \tag{2}$$

The quantity P_w can be interpreted as the probability that an information (WWW) server will be found at an arbitrarily chosen IP address A_n. In the sequel, we will refer to P_w as the information server density.

The procedure outlined above is impractical because it requires probing approximately four billion (2^{32}) addresses. Nevertheless, it is a useful starting point for our discussion and is easily made practical in the form of Monte Carlo sampling.

A. Monte Carlo Sampling

In the Monte Carlo approach, we sample only a randomly chosen subset of the Internet Protocol Version 4 (IPv4) address space. Specifically, a subset of $N_{\rm mc}$ IP addresses is chosen uniformly from the space of all 2^{32} addresses. Then each of the selected addresses A_n, $n = 1, \ldots, N_{\rm mc}$ is probed to obtain the value of the indicator function $I(A_n)$. The Monte Carlo estimator for the probability P_w is given by

$$\hat{P}_{\rm mc} = \frac{1}{N_{\rm mc}} \sum_{n=1}^{N_{\rm mc}} I(A_n). \tag{3}$$

The Monte Carlo estimator is well known and easily shown to be unbiased, i.e., $E(\hat{P}_{\rm mc}) = P_w$. Its variance equals

$$\text{var}(\hat{P}_{\rm mc}) = \frac{1}{N_{\rm mc}} \left(P_w - P_w^2\right). \tag{4}$$

It is robust and easy to implement. However, it requires large set of samples for a reliable estimate of low-probability events. It is well known that the number of samples required to achieve a given confidence interval and a given confidence level is inversely proportional to P_w. For example, in the current Internet, P_w is approximately equal to 0.2% (for the WWW service). That implies, the Monte Carlo approach requires approximately 210 000 trials to estimate P_w with a 95% confidence interval of $[0.9P_w, 1.1P_w]$.

For IPv6, the next-generation Internet protocol, this problem becomes much worse. Internet Protocol Version 6 (IPv6) fixes the problem of the limited number of available IPv4 addresses by introducing 128 bit addresses. Consequently, P_w will be on the order of 10^{-30}, and in excess of 10^{33} trials are required for reliable estimates, which makes Monte Carlo sampling completely impractical.

B. Importance Sampling

For measuring the size of the current Internet more efficiently, we propose an approach based on importance sampling to reduce significantly the sample size for a given estimation accuracy. Importance sampling is a well-known variance-reduction technique for accurately estimating the probability of rare events [3]–[5]. The principle of importance sampling is to make "interesting" events occur more frequently. This is achieved by biasing the underlying sampling density so that the events of interest have increased probability while others have reduced probability. An unbiased estimate is obtained by weighting the outcomes appropriately.

Research to date has most widely developed importance sampling for problems with continuous random variables such as the application to the estimation of error probabilities for high-performance digital communications or detection system [6]–[9], but rarely for discrete event system as in our case. Also, importance sampling has been used traditionally for simulations where all relevant statistics are known and controllable [10]. However, in our problem the underlying statistics are unknown.

Specifically, instead of uniformly selecting IP addresses as in Monte Carlo sampling, we draw independent IP addresses A_n^* to be probed from a nonuniform biasing distribution $p^*(A_n)$. The choice of this biasing distribution is central to our approach and will be discussed in detail in the next section. As we will see, this biasing distribution depends on the unknown (and not practically obtainable) *true* probability distribution $p(A_n)$, i.e., the probability that $I(A_n) = 1$.

In order to obtain an unbiased estimate, a weighting function $w(\cdot)$ is applied to the estimator. Specifically

$$\hat{P}_{\rm is} = \frac{1}{N_{\rm is}} \sum_{n=1}^{N_{\rm is}} w\left(A_n^*\right) I\left(A_n^*\right) \tag{5}$$

where N_{is} is the number of addresses probed. As long as we choose a biasing distribution $p^*(A_n)$ that is absolutely

continuous with respect to the *true* distribution $p(A_n)$, i.e., $p^*(A_n) > 0$ whenever $p(A_n) > 0$, then the weighting function $w(A_n^*) = (p_u(A_n^*)/p^*(A_n^*))$, guarantees that the estimator $\hat{P}_{is}$ is unbiased. Here, $p_u(A_n)$ denotes the uniform distribution ($p_u(A_n) = 2^{-32}$ for all addresses A_n in the address space). Recall that samples are drawn from this uniform distribution for the Monte Carlo sampling approach.

The variance of the importance sampling estimator is given by

$$\text{var}(\hat{P}_{\text{is}}) = \frac{1}{N_{\text{is}}}\left(\overline{W} - P_w^2\right) \tag{6}$$

where the average weight $\overline{W} = E_*[w^2(A^*)I(A^*)]$. Throughout, the notation $E_*[\cdot]$ denotes that the expectation is to be taken with respect to the biasing distribution $p^*(A_n)$.

Importance sampling is intended to reduce the variance of the estimator. This decreases the sampling time for a given level of accuracy, or improves the estimator accuracy for a given limited number of samples. The performance of the importance sampling estimator depends on the choice of the biasing distribution $p^*(A_n)$ and is measured by the gain γ, defined as the ratio of the "cost" of the Monte Carlo sampling estimator to that of the importance sampling estimator. Specifically, the gain is expressed as the ratio of the number of trials for a given variance or, equivalently, as the ratio of the variances for a fixed number of probes and can be expressed as

$$\gamma = \left.\frac{N_{\text{mc}}}{N_{\text{is}}}\right|_{\text{Var}_{\text{mc}}=\text{Var}_{\text{is}}} = \left.\frac{\text{Var}_{\text{mc}}}{\text{Var}_{\text{is}}}\right|_{N_{\text{mc}}=N_{\text{is}}} \tag{7}$$

$$= \frac{P_w - P_w^2}{\overline{W} - P_w^2}. \tag{8}$$

Note that the gain will be greater than one if the average weight $\overline{W}$ is less than the probability P_w.

Let us turn our attention now to the problem of choosing good biasing strategies, i.e., the search for biasing densities that maximize the gain γ.

III. Optimal Absolutely Continuous Biasing Strategy

The improvement provided by importance sampling is strongly influenced by the choice of the biasing distribution $p^*(A_n)$. Using Jensen's inequality, it can be shown that the unconstrained optimal biasing density $p^*_{\text{opt}}(A_n)$ is given by

$$p^*_{\text{opt}}(A_n) = \frac{p_u(A_n)I(A_n)}{P_w} = \frac{I(A_n)}{N_w}. \tag{9}$$

It is easily verified that this biasing density results in a perfect "estimate" of the density P_w even with only a single probe $I(A_n)$. Unfortunately, this solution is trivial and not practical because it assumes knowledge of P_w that we wish to estimate and *a priori* knowledge of the function $I(A_n)$.

However, (9) provides some useful insights. One interpretation of $p^*_{\text{opt}}(A_n)$ suggests that a good biasing strategy is to concentrate the probability mass in areas that are "promising" in the sense that they are more likely to yield a "hit." This observation leads us to introduce the thresholded biasing strategy for our probing system discussed in Section IV.

More importantly, we can interpret $p^*_{\text{opt}}(A_n)$ as the aforementioned *true* distribution $p(A_n)$, i.e., the probability of finding a web server at address A_n. To be concrete, $p(A_n)$ is given by

$$p(A_n) = \begin{cases} \frac{1}{N_w}, & \text{if } I(A_n) = 1 \\ 0, & \text{if } I(A_n) = 0 \end{cases}. \tag{10}$$

This probability distribution is unknown and cannot be obtained without probing the entire IP address space. In the sequel, we will seek to approximate marginals of this distribution to guide us in the design of good importance sampling strategies; these marginals will be referred to as empirical distributions.

A. Empirical Distributions

The *true* probability distribution $p(A_n)$ plays an important role in the design of importance sampling strategies. We have already seen that the (impractical) optimal biasing density is equal to $p(A_n)$, and we will demonstrate shortly that the gain γ of any importance sampling strategy depends on $p(A_n)$. Since we cannot obtain the complete probability distribution $p(A_n)$ itself, we will instead obtain marginals of this distribution.

These marginals capture the statistics of groups of addresses rather than individual addresses. A number of approaches exist to form such groups of addresses. We could take clues from the topology of the Internet by grouping sets of IP prefixes associated with autonomous systems. Alternatively, we could try to extract relevant groups from the way IP addresses are allocated. Instead, for this paper, we will group IP address using the conventional 4-byte description of IP addresses. It is well conceivable that one of the other approaches would lead to even better importance sampling strategies than our partitioning of the IP address space, and we feel that this is a promising area for future research. Given the paper's emphasis on the use of importance sampling, however, we believe that the use of the 4-byte description is adequate and simple.

In essence, we are aiming to bootstrap the importance sampling procedure by finding marginals of the *true* distribution of server addresses. Deriving optimal biasing strategies from these marginals is the subject of the next sections. First, let us discuss briefly how we obtained the required marginal distributions.

Let us begin by making explicit how our marginal distributions are defined. Let $p_j(b)$, $j = 1, \ldots, 4$ and $b = 0 \ldots, 255$ denote the probability of getting a positive response given that an address A_n was probed whose jth byte equals b. Then, $p_j(b)$ is related to $p(A_n)$ via

$$p_j(b) = \sum_{\substack{A_n \\ j\text{th byte equals } b}} p(A_n). \tag{11}$$

Similarly, we can form joint probabilities $p_{jk}(b_1, b_2)$ for the event that the jth byte equals b_1 and the kth byte equals b_2. We still cannot obtain the needed marginals from (11); instead, we must estimate the marginal distributions to bootstrap our procedures.

Again, we have several choices. An obvious possibility is to use uniform random (Monte Carlo) sampling to estimate the marginals. This would defeat the purpose, however, as it would require a significant number of probes before we could even

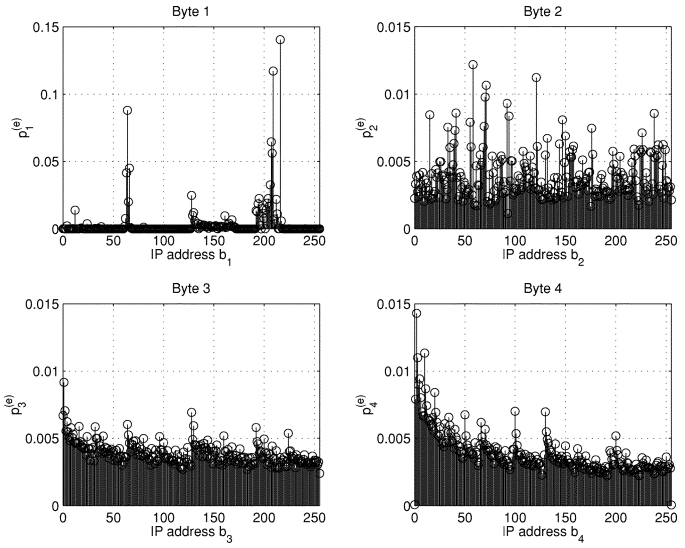


Fig. 1. Empirical distributions of each byte for number of web server addresses.

start to use importance sampling. A better approach would be to exploit some knowledge of either the IP address space topology or the mechanisms used to assign IP addresses. We opted to estimate the marginal probabilities $p_j(b)$ from a large collection of known web server addresses.

Specifically, we collected several thousand IP addresses of web servers provided by the random URL service provided by Web Crawler (http://www.webcrawler.com). These addresses were then used to form the following probability distributions, which we call empirical distributions

$$p_j^{(e)}(b) = \frac{\text{number of addresses with } j\text{th byte equal to } b}{\text{total number of collected addresses}}. \quad (12)$$

The results are depicted in Fig. 1. Patterns are discernible in particular for bytes 1 and 4. The first byte captures the consequences of how IP addresses are allocated. There are large numbers of web servers in the relatively small "class C" address range. Significantly fewer servers are present in the "class B" and "class A" ranges. Obviously, no servers are found in the reserved address ranges. The fourth bytes reflects patterns that arise from common network administration practices. For example, web servers are more likely to be assigned a forth byte with a relatively small value. No strong patterns are apparent for bytes 2 and 3.

These observations allow two conclusions. First, the fact that these patterns are explainable makes it plausible that other methods may be just as effective (or perhaps even more effective) for estimating the marginal distributions. Second, the fact that these distributions (in particular bytes 1 and 4) are not uniform will allow us to design effective importance sampling strategies.

We have conducted fairly extensive statistical analysis on the collected addresses [11]. Beyond the first-order distributions, we have focused on the question if bytes may be modeled as independent. For this purpose, we computed the mutual information $I(b_i, b_j)$, a measure of the amount of information that

TABLE I
MUTUAL INFORMATION $I(b_i, b_j)$ FOR BYTE PAIR (b_i, b_j). NUMBER OF SAMPLES = 440 000

$I(b_i, b_j)$	b_1	b_2	b_3	b_4
b_1	4.875846	1.376377	0.201028	0.062290
b_2	1.376377	7.872459	0.731200	0.121200
b_3	0.201028	0.731200	7.969250	0.106067
b_4	0.062290	0.121200	0.106067	7.890276

one random variable b_i contains about another random variable b_j [12] for all byte pairs (b_i, b_j). The mutual information is given by

$$I(b_i, b_j) = \sum_{\alpha=0}^{255} \sum_{\beta=0}^{255} p_{ij}^{(e)}(\alpha, \beta) \log_2 \frac{p_{ij}^{(e)}(\alpha, \beta)}{p_i^{(e)}(\alpha) p_j^{(e)}(\beta)} \quad (13)$$

where $p_{ij}^{(e)}$ are the joint empirical distributions for the byte pair (b_i, b_j). Small values of the mutual information indicate a low degree of dependence.

Table I lists the values $I(b_i, b_j)$ for all pairs of bytes. As a reference, we also measured the "self-information" (entropy) $I(b_i)$, given by

$$I(b_i) = \sum_{\substack{\alpha=0 \\ \alpha \in b_i}}^{255} p_i^{(e)}(\alpha) \log_2 p_i^{(e)}(\alpha)^{-1}. \quad (14)$$

These are listed on the diagonal in Table I.

We observe that the off-diagonal terms in Table I are generally much smaller than those on the diagonal. The possible exception to this statement is the pair (b_1, b_2) which may be explainable by the way IP addresses are assigned. Also, noticeable is relatively small entropy of byte 1, which reflects the strongly nonuniform distribution of that byte. We conclude that different bytes in our collected addresses show little dependence, and we will proceed to model them as independent. The slight inaccuracy of this assumption is not critical for the performance of our importance sampling strategy and this assumption simplifies our exposition and analysis greatly.

Let us return now to the problem of estimating (or approximating) the distributions $p_j(b)$. The empirical distributions $p_j^{(e)}(b)$ can be expected to accurately reflect the relative distribution of the number of web servers as a function of the byte values. Hence, we will approximate

$$p_j(b) \approx p_j^{(e)}(b) \quad (15)$$

and proceed as if this approximation holds with equality.

Furthermore, since we have determined that the empirical distributions are approximately independent across byte boundaries, we will assume that the *true* distributions are also independent for pairs of bytes. Hence, we will focus on biasing distributions with independent bytes, i.e., on biasing distributions of the form

$$p^*(A^*) = p^*(A^* = b_1.b_2.b_3.b_4) = \prod_{j=1}^{4} p_j^*(b_j). \quad (16)$$

Thus, the importance sampling weights become

$$w(A^*) = w(A^* = b_1.b_2.b_3.b_4) = \prod_{j=1}^{4} w_j(b_j) \tag{17}$$

$$= \frac{1}{2^{32}} \prod_{j=1}^{4} \frac{1}{p_j^*(b_j)}. \tag{18}$$

We are now in position to derive optimal biasing strategies for importance sampling.

B. Optimal Biasing Density

Recall that our objective is to maximize the gain of the importance sampling strategy. From (8), it is apparent that the only term in the expression for the gain γ is the average weight $\overline{W}$. In other words, the impact of choosing a particular biasing density p^* is completely represented by the functional $\overline{W}$. Hence, the optimal biasing strategy that maximizes the gain can be obtained by minimizing $\overline{W}$ via an appropriate choice of p^*.

We can compute the average weight $\overline{W}$ as follows:

$$\begin{aligned}\overline{W} &= E_* \left[w^2(A^*) I(A^*) \right] \\ &= \sum_{A_n^*} \left(\frac{p_u(A_n^*)}{p^*(A_n^*)} \right)^2 I(A_n^*) p^*(A_n^*) \\ &= P_w \sum_{A_n^*} \frac{p_u(A_n^*)}{p^*(A_n^*)} \frac{p_u(A_n^*) I(A_n^*)}{P_w}. \end{aligned} \tag{19}$$

Now, since $P_w^{-1} p_u(A_n^*) I(A_n^*)$ equals $p(A_n^*)$ [see (9)], it follows that

$$\overline{W} = P_w \sum_{A_n^*} w(A_n^*) p(A_n^*). \tag{20}$$

In order to replace the *true* distribution $p(A_n^*)$ with the empirical distribution $p^{(e)}(A_n^*)$, we invoke (15) and the independence across bytes, which leads to

$$p(A_n^*) = \prod_{j=1}^{4} p_j^{(e)}(b_j). \tag{21}$$

Inserting the last expression in (20) yields

$$\begin{aligned}\overline{W} &= P_w \prod_{j=1}^{4} \sum_{b_j=0}^{255} w_j(b_j) p_j^{(e)}(b_j) \\ &= \frac{P_w}{2^{32}} \prod_{j=1}^{4} \sum_{b_j=0}^{255} \frac{p_j^{(e)}(b_j)}{p_j^*(b_j)}. \end{aligned} \tag{22}$$

The procedure reflected in this expression is called *multibyte biasing*.

If we only bias the jth byte of the IP address distribution, and keep the other byte distributions uniform, i.e., $p_j^* \neq p_u$ and $p_i^* = p_u = (1/256)$ for $i \neq j$, then (22) specializes to

$$\overline{W}_j = P_w \sum_{b=0}^{255} w_j(b) p_j^{(e)}(b) = \frac{P_w}{256} \sum_{b=0}^{255} \frac{p_j^{(e)}(b)}{p_j^*(b)}. \tag{23}$$

This is called *single-byte biasing*.

Note, the division by $p_j^*(b)$ is uncritical as we have confined our attention to biasing strategies which are absolutely continuous with $p_j^{(e)}$. Hence, $p_j^*(b)$ can only be zero if $p_j^{(e)}$ is also zero. In that case, the ratio of the two probabilities is taken to equal zero.

Expressions (22) and (23) make explicit the dependence of the average weight $\overline{W}$ on the biasing density. They form the starting point for the design of an optimal biasing strategy. The optimal single byte biasing strategy can be found by Lagrangian optimization of $\overline{W}_j$ as shown in the following theorem.

Theorem 1: Among all possible biasing strategies leading to an unbiased estimate of P_w, the biasing density

$$\widetilde{p}_j^*(b) = \frac{\sqrt{p_j^{(e)}(b)}}{\sum_{b_k=0}^{255} \sqrt{p_j^{(e)}(b_k)}}, \quad b = 0, \ldots, 255 \tag{24}$$

is the single byte biasing strategy that maximizes the gain γ.

Proof: The optimal biasing strategy can be found by minimizing $\overline{W}_j$. Let the Lagrangian objective function be

$$J = \sum_{b=0}^{255} \frac{p_j^{(e)}(b)}{p_j^*(b)} + \lambda \left(\sum_{b=0}^{255} p_j^*(b) - 1 \right).$$

For each address b, differentiating with respect to $p_j^*(b)$ and setting the result equal to zero yields $p_j^*(b) = \sqrt{(p_j^{(e)}(b)/\lambda)}$. The constraint requires $\sum_{b=0}^{255} p_j^*(b) = 1$, which gives $\lambda = (\sum_{b=0}^{255} \sqrt{p_j^{(e)}(b)})^2$ and (24) results. Since $\nabla^2 J(p_j^*(b)) \geq 0$, $p_j^*(\cdot)$ is the optimal biasing density for jth byte.

An unbiased estimate is obtained since the underlying sample distribution is absolutely continuous with respect to this constrained optimal biasing distribution. ■

For multibyte biasing, (22) and (23) yield

$$\overline{W} = P_w \prod_{j=1}^{4} \frac{\overline{W}_j}{P_w} \tag{25}$$

which implies that applying the optimal single byte biasing strategy to each byte individually will lead to optimal multibyte biasing. Further gain can be realized from this strategy, since

$$\gamma \approx \frac{P_w}{\overline{W}} = \prod_{j=1}^{4} \frac{P_w}{\overline{W}_j} \approx \prod_{j=1}^{4} \gamma_j \tag{26}$$

where γ_j is the gain obtained by biasing the distribution of the jth byte.

C. Experiments

We experimented with the optimal single byte strategy and found that this biasing scheme is nearly seven times more efficient than Monte Carlo sampling for an unbiased estimate of the web server density P_w. More specifically, to obtain a reliable estimate of P_w, Monte Carlo sampling needs to probe more than 200 000 IP addresses (refer to Section II-A). Put differently, if Monte Carlo sampling would require a week to complete, we could obtain an estimate with the same accuracy in a single day using importance sampling.

The results of a sampling run are illustrated in Fig. 2. The curves in the top figure show the "running" estimates using both

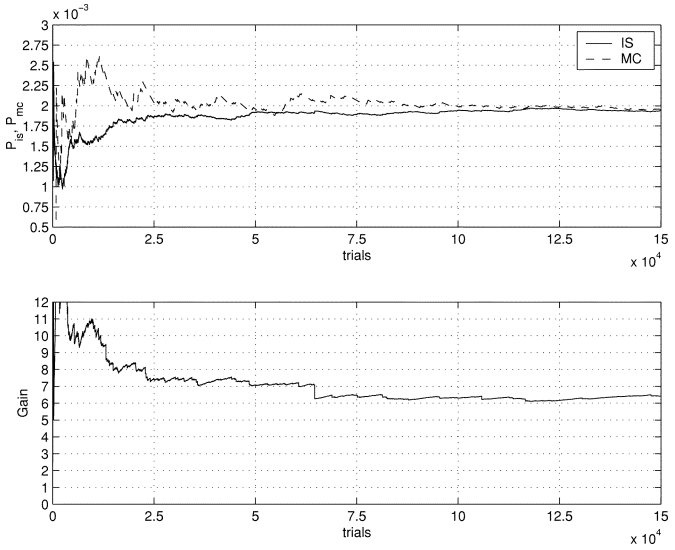


Fig. 2. Optimal byte 1 biasing importance sampling versus Monte Carlo estimation of web server density. Top: density of web servers. Bottom: gain of the estimator. Test date: December 11, 2000.

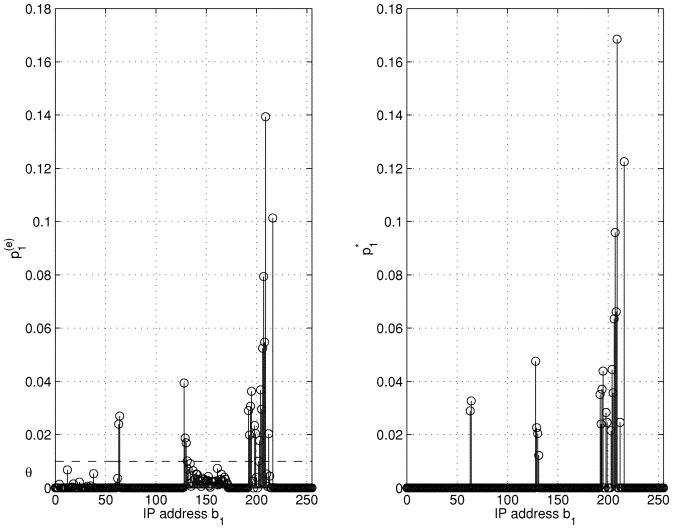


Fig. 3. Empirical versus biasing distribution for byte 1 of the IP addresses of web servers based on threshold approach. Left: original. Right: thresholded.

Monte Carlo and importance sampling. After each probe, the respective estimates are updated and plotted. The importance sampling estimate settles clearly faster than the Monte Carlo estimate reflecting the reduced variance. The experiment was conducted on December 11, 2000, and estimates P_w to equal 0.2% which corresponds to approximately 8.3 million publicly accessible web servers. An estimate of the gain of the importance sampling estimator over Monte Carlo sampling is shown in the bottom figure.

We conclude this section by noting that the savings achieved by our constrained optimal importance sampling strategy over the Monte Carlo approach are modest. This limited gain results from the absolute continuity condition applied to biasing distributions $p^*(\cdot)$. No parametric statistic models for the distribution of web servers over the IP address space are available. This makes the entire distribution (rather than only a few parameters for a parametric system) candidates for modification toward a global optimum. Hence, the optimal importance sampling strategy for our system depends highly on the underlying distribution. Therefore, it is extremely difficult to obtain high gain without relaxing the absolute continuity condition in the "nonpromising" regions which result in a low "hit" rate.

Furthermore, for our problem, the effectiveness of the biasing strategy depends on $p_j^{(e)}$. It can be shown that the importance sampling approach with single byte biasing is most effective if

$$p_j^{(e)}(b) = \begin{cases} 1, & \text{for some } b \\ 0, & \text{for all other } b\text{'s} \end{cases} \tag{27}$$

which leads to the minimum average weight $\overline{W}_{\min} = (P_w/256)$ and maximum gain $\gamma_{\max} \approx (P_w/\overline{W}) = 256$.

Hence, it implies that the importance sampling strategy, in general, is more effective if there are many zeros in the *true* distribution of server IP address p_j. This observation leads us to devise more efficient estimators as shown in the next section. It also indicates that we may expect much higher gains with IPv6, as the occupation of addresses will be much sparser in the IPv6 address space.

IV. Higher Gain via Thresholding

The gain for unbiased estimates is limited by the absolute continuity condition for the biasing distribution p^*. To speedup the convergence of the estimate and increase the gain further, we consider biasing schemes that introduce some known or estimable bias. This strategy aims to increase the gain by drawing more samples from "important areas" as discussed for the unconstrained optimal biasing strategy [(9)] or, equivalently, creating more zeros in the biasing distribution as discussed in Section III-C.

One possible approach is to shrink the "promising" sample set by setting an appropriate threshold θ in the empirical distribution for IP addresses of information servers [13]. Specifically, for the jth byte of an IP address, define

$$\tilde{a}_j(b) = \begin{cases} p_j^{(e)}(b), & \text{if } p_j^{(e)}(b) > \theta_j \\ 0, & \text{otherwise} \end{cases} \tag{28}$$

for $b = 0, \cdots, 255$. Then, the biasing distribution is given by

$$p_j^*(b) = \frac{\tilde{a}_j(b)}{\sum_{b=0}^{255} \tilde{a}_j(b)}, \quad \text{for } b = 0, \ldots, 255. \tag{29}$$

Fig. 3 illustrates the transformation from the original empirical distribution into the thresholded biasing distribution for byte 1 of IP addresses of web servers. The samples of IP addresses with probability less than the threshold will be no longer probed, while more samples in "important areas" indicated by a high probability will be drawn. The primary intuition behind this approach is that more positive responses will be created during the probing of information servers. Hence, the importance sampling strategy is more effective and an increase of the estimator gain follows.

A. Underestimation and Correction

As we mentioned in the beginning of this section, the thresholded biasing strategy introduces a biased estimate, since the threshold biasing distribution will not be absolutely continuous with respect to underlying sample distribution. However, the

bias is quantifiable as shown in the following theorem and we will propose below means for estimating the bias.

Theorem 2: Let us denote by B the bias factor $B = \prod_{j=1}^{4} B_j$, where

$$B_j = \sum_{\substack{b=0 \\ p_j^{(e)}(b) \geq \theta_j}}^{255} p_j^{(e)}(b) \tag{30}$$

is the bias factor for the jth byte of IP addresses. The expectation of the estimator $\hat{P}^{(\mathrm{th})}$ for estimating P_w obtained with the thresholded biasing density is given by

$$\begin{aligned} E\left(\hat{P}^{(\mathrm{th})}\right) &= P_w \prod_{j=1}^{4} \sum_{\substack{b=0 \\ p_j^{(e)}(b) \geq \theta_j}}^{255} p_j^{(e)}(b) \\ &= P_w \prod_{j=1}^{4} B_j = P_w B. \end{aligned} \tag{31}$$

Proof: The jth byte weighting function for the threshold approach is $w_j(b_j) = (1/256 p_j^*(b_j))$ if $p_j^{(e)}(b_j) \geq \theta_j$ and $w_j(b_j) = 0$, otherwise, for $b_j = 0, \ldots, 255$. Then

$$\begin{aligned} E\left(\hat{P}^{(\mathrm{th})}\right) &= \frac{1}{N_{is}} \sum_{n=1}^{N_{is}} E_* \left[w\left(A_n^*\right) I\left(A_n^*\right) \right] \\ &= \sum_{\substack{A_n^* \\ p_j^{(e)}(b_j) \geq \theta_j,\ j=1\ldots4}} \frac{p_u\left(A_n^*\right)}{p^*\left(A_n^*\right)} I\left(A_n^*\right) p^*\left(A_n^*\right) \\ &= P_w \sum_{\substack{A_n^* \\ p_j^{(e)}(b_j) \geq \theta_j,\ j=1\ldots4}} \frac{p_u\left(A_n^*\right) I\left(A_n^*\right)}{P_w} \\ &= P_w \prod_{j=1}^{4} \sum_{\substack{b=0 \\ p_j^{(e)}(b) \geq \theta_j}}^{255} p_j^{(e)}(b) \end{aligned} \tag{32}$$

where the last equation follows from (21) and independence. Equation (31) follows from this argument. Since $E(\hat{P}_{\mathrm{unbiased}}) = P_w$, the relative bias δ resulting from the cumulative probability of discarded samples, can be shown to equal

$$\delta = \frac{E(\hat{P}_{\mathrm{unbiased}}) - E\left(\hat{P}^{(\mathrm{th})}\right)}{E(\hat{P}_{\mathrm{unbiased}})} \tag{33}$$

$$= 1 - B \tag{34}$$

$$= 1 - \prod_{j=1}^{4} \left(1 - \sum_{\substack{b=0 \\ p_j^{(e)}(b) < \theta_j}}^{255} p_j^{(e)}(b) \right), \quad 0 \leq \delta \leq 1. \tag{35}$$

■

It follows immediately that underestimation occurs as the threshold estimator is always biased to a smaller value than the true value due to the omitted addresses with small "hit" rates. However, an unbiased estimate $\hat{P}_{\mathrm{unbiased}}$ can be obtained if the relative bias δ or, equivalently, the bias factor B is known or estimable via (31), i.e.,

$$P_w = E(\hat{P}_{\mathrm{unbiased}}) = \frac{E\left(\hat{P}^{(\mathrm{th})}\right)}{B}. \tag{36}$$

Hence, an implementable threshold biasing strategy depends on estimating the bias factor B.

Equation (30) provides the basis for estimating the bias factor B via the empirical data. For measuring the density of web servers, for example, we may calculate B independently from the empirical distributions for the IP addresses extracted from several thousand random URLs introduced in Section III-A. However, there are no such databases available for services other than WWW, such as telnet, FTP, sendmail, etc. It may be possible to obtain equivalent expressions when marginal distributions are obtained by other methods. An alternative approach for estimating B will be proposed later in Section V.

B. Effectiveness of Thresholded Biasing

The resulting variance of the thresholded estimator is

$$\mathrm{var}\left(\hat{P}^{(\mathrm{th})}\right) = \frac{1}{N_{\mathrm{is}}} \left(\overline{W}^{(\mathrm{th})} - P_w^2 B^2 \right) \tag{37}$$

where the average weight $\overline{W}^{(\mathrm{th})} = E_*[(w^{(\mathrm{th})}(A_n^*))^2 I(A_n^*)]$. Then, the estimator gain will be

$$\gamma^{(\mathrm{th})} = \frac{P_w - P_w^2}{\overline{W}^{(\mathrm{th})} - P_w^2 B^2}. \tag{38}$$

Clearly, the gain for the threshold estimator is mostly determined by $\overline{W}^{(\mathrm{th})}$. Let us denote by Q the number of IP addresses for which the empirical distribution $p_j^{(e)}(b)$ of the jth byte is greater than threshold θ_j. Then, for single byte biasing, by (23) and (29), $\overline{W}_j^{(\mathrm{th})}$ will be

$$\overline{W}_j^{(\mathrm{th})} = \frac{Q P_w}{256} \sum_{\substack{b=0 \\ p_j^{(e)}(b) \geq \theta_j}}^{255} p_j^{(e)}(b) = \frac{Q}{256} P_w B_j. \tag{39}$$

A significant gain is achieved with single byte thresholding, given approximately by

$$\gamma_j^{(\mathrm{th})} \approx \frac{P_w}{\overline{W}_j^{(\mathrm{th})}} = \frac{256}{Q B_j} \gg 1 \tag{40}$$

since $Q < 256$ (nonuniform empirical distributions), and $B_j \leq 1$.

If $\theta_j = 0$, i.e., the jth byte biasing density is chosen to be the empirical distributions $p_j^{(e)}$, then $B_j = 1$, and a gain $(256/Q) > 1$ is still obtained with an unbiased estimate.

Further, the higher the threshold, the more zeros the biasing distributions will have. Hence, $\overline{W}_j^{(\mathrm{th})}$ has a much smaller value, resulting in a much smaller variance of the biased estimator. It can be shown easily that the thresholded importance sampling estimator with single byte biasing is most effective if

$$p_j^*(b) = \begin{cases} 1, & \text{for } b = \arg\max \left\{ p_j^{(e)}(b) \right\} \\ 0, & \text{for all other } b's \end{cases}. \tag{41}$$

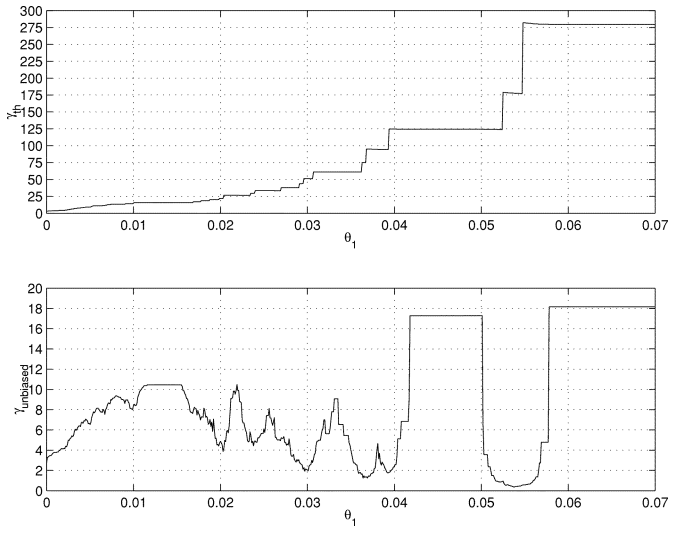


Fig. 4. Predicted biased/unbiased gain versus threshold for the biasing distribution of byte 1 of web server IP addresses. $P_w = 0.001\,764$, Number of trials $= 150\,000$. Top: biased gain. Bottom: unbiased gain.

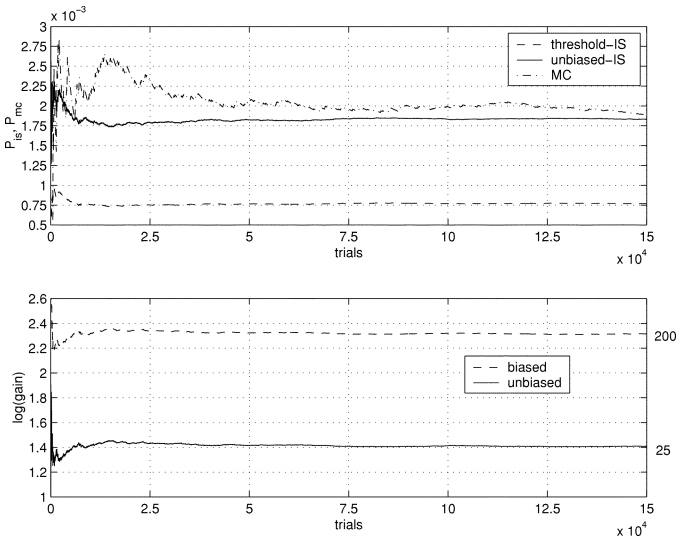


Fig. 5. Threshold importance sampling versus Monte Carlo estimation of web server density. $\theta_1 = 0.048$. Top: density of web servers. Bottom: logarithmic gain of the estimator. Test date: November 30, 2000.

For multibyte biasing, an appropriate threshold will be set for each empirical byte distribution. Then, the gain is significant since $\gamma^{(\mathrm{th})} = \prod_{j=1}^{4} \gamma_j^{(\mathrm{th})}$, where $\gamma_j^{(\mathrm{th})}$ is the gain for biasing the jth byte.

C. Choosing the Threshold

Biasing via thresholding achieves a significantly improved estimate over the absolutely continuous biasing strategies. This is achieved by additional work due to the need to correct the bias. Note that the bias factor B is a random variable. The variance of B must be reflected in the variance of the unbiased estimate, $\mathrm{var}(\hat{P}_{\mathrm{unbiased}})$, which can be derived as

$$\begin{aligned}\mathrm{var}(\hat{P}_{\mathrm{unbiased}}) = &\mathrm{var}\left(\hat{P}^{(\mathrm{th})}\right)\cdot \mathrm{var}\left(\frac{1}{B}\right)\\ &+ E^2\left(\hat{P}^{(\mathrm{th})}\right)\cdot \mathrm{var}\left(\frac{1}{B}\right)\\ &+ E^2\left(\frac{1}{B}\right)\cdot \mathrm{var}\left(\hat{P}^{(\mathrm{th})}\right). \qquad (42)\end{aligned}$$

In the case that the empirical data is available, the expectation and variance of random variable $1/B$ may be calculated via the collected IP addresses of information servers. And, the variance of $1/B$ can be reduced by averaging a larger collection of empirical samples.

From (42), we see that $\mathrm{var}(\hat{P}_{\mathrm{unbiased}})$ is greater than $\mathrm{var}(\hat{P}^{(\mathrm{th})})$ since $E(1/B) \geq 1$. However, a very high unbiased gain can still be achieved. The predicted biased and unbiased gain versus threshold for the biasing distribution for the first byte of IP addresses of web servers is shown in Fig. 4, which provides a basis for the considerable tradeoff between the bias and the unbiased gain.

Fig. 4 shows that the reduction in trials is almost exponential over the threshold. The flat part in the curves indicates no distribution of server address falls into that threshold interval, giving a consistent biased and unbiased gain. The notches in the unbiased gain curve result from the jump in the variance of bias in that threshold interval, which implies that for those empirical distribution values around the threshold, called "sensitive distributions," their variance has a significant effect on the variance of the bias, thus, increasing the entire variance of the unbiased estimate.

Hence, a near-optimal threshold should be set to avoid those "sensitive distributions" and correspond to an estimator whose unbiased gain is high and both the biased and unbiased gain fall in the flat region of the pattern for a robust estimate.

Fig. 5 illustrates the results of our experiments comparing importance sampling and Monte Carlo sampling estimate to the web server density for a threshold of 0.048 in the empirical distribution of byte 1 of the IP address. We can see that a biased threshold estimate achieves a biased gain of 200. However, the unbiased estimate result is corrected and a unbiased gain of 25 is obtained.

V. Estimating the Bias Without Empirical Distributions

The threshold estimator $\hat{P}^{(\mathrm{th})}$ is a special case of the general biased estimator $\hat{P}_{\mathrm{biased}}$ for estimating the density P_w of an information server. For biasing the jth byte of an IP address

$$E(\hat{P}_{\mathrm{biased}}) = P_w \sum_{\substack{b=0\\ b\in F}}^{255} p_j^{(e)}(b). \qquad (43)$$

The biasing distribution of $\hat{P}_{\mathrm{biased}}$ is given by

$$p_j^*(b) = \begin{cases} p_j^F(b), & b \in F \\ 0, & \text{otherwise} \end{cases}. \qquad (44)$$

Here, F is a collection of promising samples which lead to "hits" frequently, hence resulting in a high-performance biased estimator. For the threshold approach, $F = \{b | p_j^{(e)}(b) \geq \theta_j, b = 0, \ldots, 255\}$, where θ_j is the threshold set for the jth byte, and p_j^F is given by (29).

An unbiased estimate of P_w is obtained by correcting the bias B_j, i.e., $E(\hat{P}_{\mathrm{unbiased}}) = P_w = (E(\hat{P}_{\mathrm{biased}})/B_j)$, where $B_j = \sum_{\substack{b=0\\ b\in F}}^{255} p_j^{(e)}(b)$. The advantage of the biased estimator $\hat{P}_{\mathrm{biased}}$ is

its high performance and generality. It can be applied to a wide range of discrete systems to achieve an unbiased estimate, if, the bias factor B_j of interest is known or estimable.

As mentioned before, empirical data for web server addresses are available. Thus, B_j may be easily estimated via the collected IP addresses of web servers. However, there are many cases where no such databases are available for services other than WWW, such as FTP, sendmail, etc. In these cases, we must estimate both the information server density P_w and the bias factor B_j via probing.

A. Estimating the Bias via Importance Sampling

A possible approach for evaluating B_j is the Monte Carlo approach. Let us denote by H the set of IP address resulting in hits during Monte Carlo trials. Then

$$B_j = P(F|H) = \frac{E\left(I(F)\right)}{E\left(I(H)\right)} = \frac{E\left(I(F)\right)}{E\left(I(F)\right) + E\left(I(F^c)\right)}. \tag{45}$$

The Monte Carlo estimator of the bias B_j is defined as

$$\hat{B}_j^{(\mathrm{mc})} = \frac{\sum_{\substack{n=1\\ b\in F}}^{L} I\left(A_n(b)\right)}{\sum_{\substack{n=1\\ b\in F}}^{L} I\left(A_n(b)\right) + \sum_{\substack{n=1\\ b\in F^c}}^{L} I\left(A_n(b)\right)} = \frac{N_F}{N_H} \tag{46}$$

where N_F and N_H are the number of hits corresponding to the address set F and H, respectively.

The Monte Carlo method provides a way to estimate the bias from the trials in the case that the offline empirical distribution $p^{(e)}$ of information servers is not available. However, it is an inefficient estimate since a large number of hits N_F is required for a reliable estimate of B_j, which will be equally computationally expensive as finding P_w itself.

To alleviate this problem, we propose to use an importance sampling-based technique that combines the biased estimator $\hat{P}_{\mathrm{biased}}$ and the Monte Carlo method. This approach enjoys the advantage of yielding a larger number of hits corresponding to B_j via $\hat{P}_{\mathrm{biased}}$, hence providing faster convergence to B_j than Monte Carlo method and providing hits corresponding to the set H via Monte Carlo sampling for a computable estimate of B_j.

The resulting single-byte biasing density over the entire address space of the jth byte will be

$$p_j^*(b) = \alpha p_j^F(b) + \frac{1-\alpha}{256}, \quad \text{for } b = 0, \ldots, 255 \tag{47}$$

where the biasing density p_j^F corresponds to a promising IP address set F of the jth byte. A more efficient estimate is achieved by the mixture factor $\alpha (0 < \alpha < 1)$. Then, an unbiased estimate of the bias B_j based on this approach is obtained and given by

$$\hat{B}_j^{(\mathrm{is})} = \frac{\sum_{\substack{n=1\\ b\in F}}^{M} I\left(A_n(b)\right) w_j(b)}{\sum_{\substack{n=1\\ b\in F}}^{M} I\left(A_n(b)\right) w_j(b) + \sum_{\substack{n=1\\ b\in F^c}}^{M} I\left(A_n(b)\right) w_j(b)} \tag{48}$$

where the weighting function $w_j(b) = (1/256 p_j^*(b))$, for $b = 0, \ldots, 255$, and we can expect that $M < L$ for a given level of accuracy for estimating B_j.

It should be pointed out that there is no *a priori* knowledge of the probed information servers (e.g., FTP servers) before the trials. Hence, the biasing density $p_j^F(\cdot)$ may be generated initially from a known empirical data set such as the empirical distributions of web servers via thresholding. This is based on the observation that the difference between the underlying statistics of two information services, such as FTP and WWW, should not be very significant.

B. Choosing the Mixture Factor

Clearly, the performance of our proposed importance sampling-based approach for estimating bias B_j is strongly influenced by the choice of α. Note that this approach will also provide an unbiased estimate of the information server density P_w. Hence, α should be chosen to obtain the maximum gain γ_j of the estimator $\hat{P}_{\mathrm{unbiased}}$.

Consider that $\hat{P}_{\mathrm{unbiased}}$ must also be composed of estimates obtained respectively from Monte Carlo trials and the biased approach with a fraction $\lambda (0 < \lambda < 1)$. Then

$$\hat{P}_{\mathrm{unbiased}} = \lambda \hat{P}_{\mathrm{mc}}\left((1-\alpha)M\right) + (1-\lambda)\frac{\hat{P}_{\mathrm{biased}}(\alpha M)}{\hat{B}_j} \tag{49}$$

where $\hat{P}_{\mathrm{mc}}(N)$ and $\hat{P}_{\mathrm{biased}}(N)$ are Monte Carlo and biased estimator with N trials, respectively.

The optimal value of λ which minimizes the variance of the estimator $\hat{P}_{\mathrm{unbiased}}$ can be easily found by Lagrangian optimization of $\hat{P}_{\mathrm{unbiased}}$, which leads to the following result:

$$\lambda_{\mathrm{opt}} = \frac{V_2}{V_1 + V_2} \tag{50}$$

where

$$\begin{aligned} V_1 &= \mathrm{var}\left(\hat{P}_{\mathrm{mc}}\left((1-\alpha)M\right)\right) \\ &= \frac{1}{(1-\alpha)M}\left(P_w - P_w^2\right) \end{aligned} \tag{51}$$

and

$$\begin{aligned} V_2 &= \mathrm{var}\left(\frac{\hat{P}_{\mathrm{biased}}(\alpha M)}{\hat{B}_j}\right) \\ &= \mathrm{var}\left(\hat{P}_{\mathrm{biased}}(\alpha M)\right)\mathrm{var}\left(\frac{1}{\hat{B}_j}\right) \\ &\quad + E^2\left(\frac{1}{\hat{B}_j}\right)\mathrm{var}\left(\hat{P}_{\mathrm{biased}}(\alpha M)\right) \\ &\quad + E^2\left(\hat{P}_{\mathrm{biased}}(\alpha M)\right)\mathrm{var}\left(\frac{1}{\hat{B}_j}\right). \end{aligned} \tag{52}$$

Thus, the gain γ_j for λ_{opt} is given by

$$\gamma_j = \left.\frac{\mathrm{var}(\hat{P}_{\mathrm{mc}})}{\mathrm{var}(\hat{P}_{\mathrm{unbiased}})}\right|_{N_{\mathrm{unbiased}}=N_{\mathrm{mc}}=M} = (1-\alpha)\left(1+\frac{V_1}{V_2}\right). \tag{53}$$

In (52), $E(\hat{P}_{\mathrm{biased}}(\cdot))$ and $\mathrm{var}(\hat{P}_{\mathrm{biased}}(\cdot))$ can be obtained via (31) and (37), respectively. Hence, a remaining problem for evaluating γ_j is how to estimate the moments of random variable $1/\hat{B}_j$.

Recall that $\hat{B}_j = (N_F/N_H)$, where N_F and N_H are the number of hits corresponding to the address set F and H, respectively. Then, the conditional moments of $1/\hat{B}_j$ given that $N_F = n_f$ will be $E((1/\hat{B}_j)|n_f) = (1/n_f)E(N_H|n_f)$ and $\mathrm{var}((1/\hat{B}_j)|n_f) = (1/n_f^2)E(N_H^2|n_f) - (1/n_f^2)E^2(N_H|n_f)$.

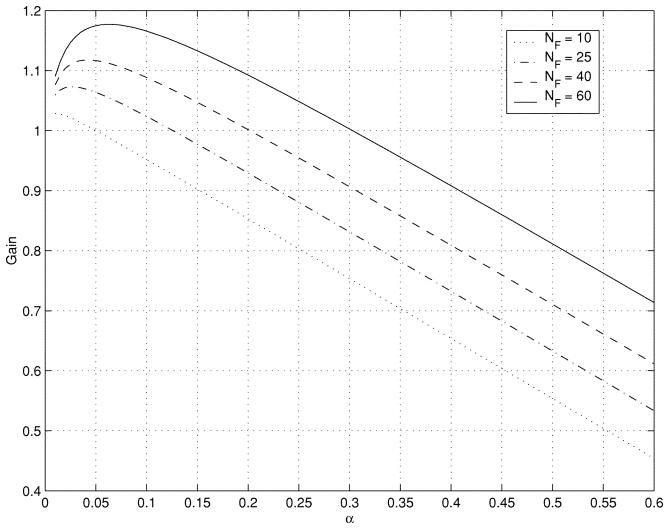


Fig. 6. Predicted gain γ_j versus α for FTP servers. $M = 140\,000$, $P_w = 0.0023$, $Q = 5$, and $B_j = 0.28$.

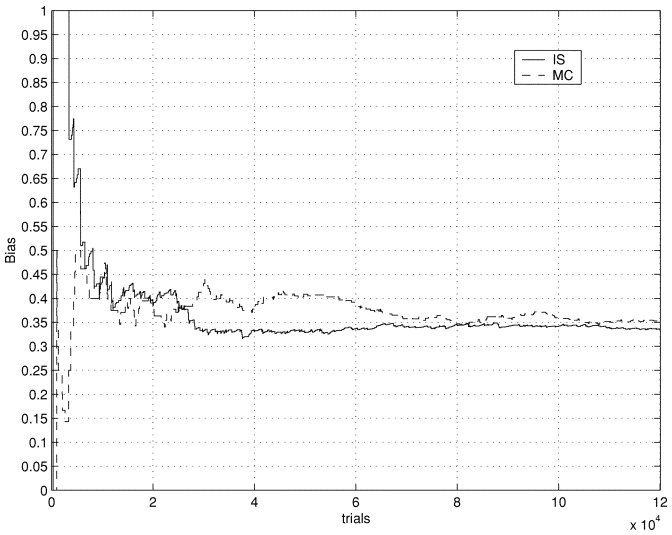


Fig. 7. Bias factor B_1 for estimating the FTP server density. Promising address set $F = [64\ 207\ 209\ 212\ 216]$, $\alpha = 0.1$. Test date: November 13, 2001.

Consider that N_F is a binomial random variable with parameters N_H and B_j. It is reasonable to approximate N_F by a Poisson arrival process for large N_H and small B_j. Furthermore, similar to the probability density function which describes the time required to observe n_f arrivals from a Poisson process [14], the conditional probability mass function (pmf) of N_H given that $N_F = n_f$ can be expressed by discretized Erlang-like distributions, given by

$$P(N_H = n_h | N_F = n_f) = \frac{B_j^{n_f} n_h^{n_f - 1}}{(n_f - 1)!} e^{-B_j n_h}. \qquad (54)$$

Then, the conditional moments $E(N_H|n_f)$ and $E(N_H^2|n_f)$ are obtained straightforwardly by the pmf $P(N_H|N_F)$, and a computable solution of the moments of $1/\hat{B}_j$ results under the condition $N_F = n_f$.

Fig. 6 illustrates the predicted gain γ_j of the estimator $\hat{P}_{\text{unbiased}}$ for FTP servers. The curve shows γ_j versus α for several given values of N_F. It provides a basis for selecting a proper mixture factor α for our importance sampling-based approach.

C. Evaluating the Bias and the Information Server Density

Note that the gain γ_j for estimating P_w on the proposed importance sampling-based approach is modest. Hence, it is not efficient to perform a long run based on this approach to obtain a reliable estimate of P_w. However, a large hit set corresponding to the promising address set F is generated through a short set of trials, resulting in a smaller variance of $\hat{B}_j^{(\text{is})}$ than the variance of $\hat{B}_j^{(\text{mc})}$.

An experiment for evaluating B_j for estimating FTP server density via the importance sampling-based approach is shown in Fig. 7. The Monte Carlo method is also shown in the figure for comparison. Each point in the figure represents an estimate obtained by probing the remote host's TCP well-known port 21 for as many times as indicated on the x axis. Responses with status code 2XX are counted as a successful request. The promising address set F is determined by thresholding a known empirical distributions of web servers. We see that the importance sampling-based approach provides more stable estimate of the bias B_j and faster convergence than the Monte Carlo method.

Hence, the importance sampling-based approach provides an alternative method for estimating the bias factor B_j without empirical distributions. Although this approach will provide simultaneously an unbiased estimate of P_w after a long set of trials, it will be made more efficient by performing a second biased importance sampling for estimating P_w only after obtaining the estimation of B_j based on a short run using this approach. It will reduce the total sampling time for estimating both B_j and P_w. Thus, an algorithm designed for estimating P_w without a prior empirical distributions of probed information servers proceeds as follows.

1) Initialize by finding a promising address set F and $p_j^F(\cdot)$, which can result from thresholding known empirical distributions of some information servers, such as web servers.
2) Probe with a short run by using the biasing density combined by $p_j^F(\cdot)$ and uniform density (Monte Carlo) [(47)]. Calculate $B_j^{(is)}$ for the address set F.
3) Run a second short set of probes with biasing density $p_j^F(\cdot)$ based on the biased importance sampling approach [(43) and (44)]. Calculate P_w via $B_j^{(\text{is})}$.

It should be pointed out that the second step will stop immediately once a large hit set corresponding to F is achieved for estimating $B_j^{(\text{is})}$ and the procedure will then switch to the third step.

Fig. 8 illustrates the third step of an experiment for estimating the FTP server density P_w. The biasing density $p_1^F(\cdot)$ in this short run is generated by thresholding a known empirical distributions of web servers. A biased estimate of P_w is introduced with a gain as high as 160 in comparison with the Monte Carlo method. An unbiased estimate $\hat{P}_{\text{unbiased}}$ is achieved by correcting the bias $B_1^{(\text{is})}$ which is obtained by a early short run using the proposed importance sampling-based approach.

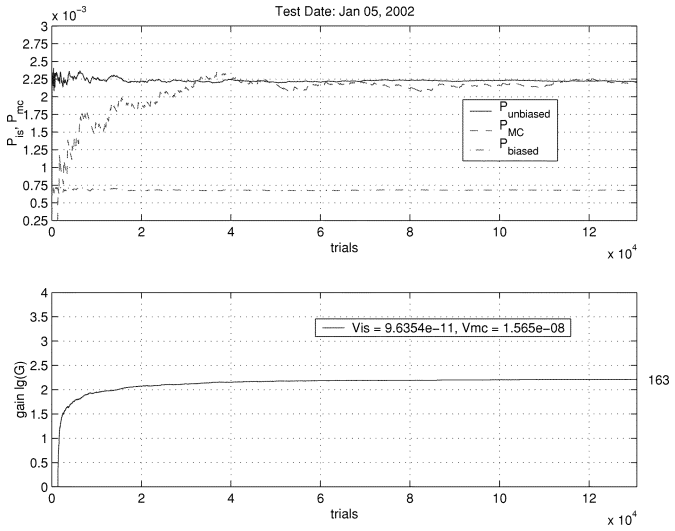


Fig. 8. Biased importance sampling versus Monte Carlo estimation of FTP server density. $F = [64\ 207\ 209\ 212\ 216]$, $B_{\text{is}}^{1} = 0.305$. Top: density of FTP servers. Bottom: logarithmic gain of the estimator. Test date: January 5, 2002.

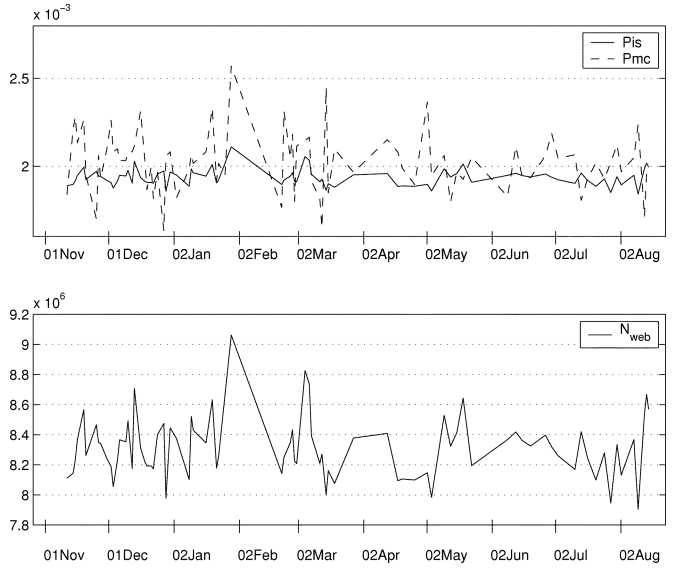


Fig. 9. Number of IP addresses with publicly accessible web servers from November 2001 to August 2002. Top: density of web servers as measured via Monte Carlo and importance sampling. Bottom: number of web servers.

VI. Measurement Results

Based on the approaches presented above, we have made periodic measurements of the prevalence of WWW services on the Internet to map the growth of the current Internet. Fig. 9 illustrates the development of the size of the Internet as measured by the number of IP addresses with publicly accessible web servers from November 2001 to August 2002. Each data point is the result of probing nearly 60 000 IP addresses. The larger variance of the results obtained through Monte Carlo sampling are clearly evident.

Surprisingly, Fig. 9 shows a nearly constant number of IP addresses with publicly accessible web servers. This observation is in stark contrast to measurements provided by, e.g., Netcraft [2] which demonstrates continued growth of the number of *domains* providing web servers. Explanations for the observed difference are not immediately obvious. One observation we have made is that if we include "negative" responses (in particular, `4XX` responses) in our tallies, then the number of IP addresses providing web services is growing. We have not been able to shed further light on this observations. Second, the difference in these measurements might be explained (at least in part) by an increasing number of web sites provided on the same host (IP address); such sites are generally referred to as *virtual hosts*. At this time, our methods do not provide means to detect multiple WWW domains operating on the same IP address.

VII. Conclusion

The Internet has been growing rapidly and substantially. Measuring the size of Internet is an important open problem and has attracted more attention recently. In this paper, an optimal importance sampling strategy has been presented, which is nearly seven times more efficient than Monte Carlo sampling for an unbiased estimate of the web server density. In order to speedup the convergence of the estimate and increase the gain more significantly, we allow biasing densities that are not absolutely continuous with respect to the actual distribution of information servers over the IP address space. The biasing densities result from thresholding empirically observed address distributions and result in very high gains. They also result in a biased estimator. The advantage of thresholded biasing strategy is its generality and applicability to a wide range of discrete systems to achieve unbiased estimate, if, the bias is known or estimable. For measuring the density of web servers, we may calculate the bias by the empirical distributions for IP addresses extracted from several thousand random URLs provided by the web crawler. In most cases such as for estimating the density of FTP or telnet servers, however, the empirical data is not available.

To combat this problem, we proposed an importance sampling-based approach which combines the estimates from Monte Carlo and biased importance sampling to estimate the bias. An algorithm designed for estimating the density of an information server without a prior empirical distributions of that server is presented. An estimate for FTP server density and the bias based on this algorithm was obtained with a significant reduction of the total sampling time.

In summary, this paper has introduced novel efficient and effective statistical methods for measuring the size of IPv4 Internet based on importance sampling. Specifically, a thorough analysis of our importance sampling scheme is performed and compared with the Monte Carlo sampling technique. An accurate estimate for the current size of the Internet has been obtained as measured by the number of publicly accessible web servers and FTP servers. This framework will be applied in the future to measure the growth dynamics of the Internet.

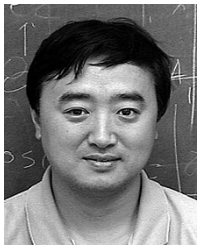
References

[1] Internet Domain Survey Background, Internet Software Consortium. [Online]. Available: http://www.isc.org/ds/new-survey.html

[2] The Netcraft Web Server Survey, Netcraft. [Online]. Available: http://www.netcraft.com/survey

[3] P. W. Glynn and D. L. Iglehart, "Importance sampling for stochastic simulations," *Manage. Sci.*, vol. 35, pp. 1367–1392, Nov. 1989.

[4] P. Heidelberger, "Fast simulation of rare events in queueing and reliability models," *ACM Trans. Mod. Comput. Simul.*, vol. 5, no. 1, pp. 43–85, Jan. 1995.
[5] K. S. Shanmugam and P. Balaban, "A modified Monte-Carlo simulation technique for the evaluation of error rate in digital communication systems," *IEEE Trans. Commun.*, vol. COM–28, pp. 1916–1924, Nov. 1980.
[6] G. C. Orsak and B. Aazhang, "On the theory of importance sampling applied to the analysis of detection systems," *IEEE Trans. Commun.*, vol. 37, pp. 332–339, Apr. 1989.
[7] S. L. Smith and G. C. Orsak, "A modified importance sampling scheme for the estimation of detection system performance," *IEEE Trans. Commun.*, vol. 43, pp. 1341–1346, Feb. 1995.
[8] D. Liu and K. Yao, "Improved importance sampling technique for efficient simulation of digital communication systems," *IEEE J. Select. Areas Commun.*, vol. 6, pp. 67–75, Jan. 1988.
[9] G. C. Orsak, "A note on estimating false alarm rates via importance sampling," *IEEE Trans. Commun.*, vol. 41, pp. 1275–1277, Sept. 1993.
[10] P. J. Smith, M. Shafi, and H. Gao, "Quick simulation: a review of importance sampling techniques in communications systems," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 597–613, May 1997.
[11] S. Xing and B.-P. Paris, "Importance sampling for measuring the size of the Internet," in *Proc. 35th Conf. Information Sciences Systems*, vol. 2, Baltimore, MD, Mar. 2001, pp. 593–597.
[12] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. New York: McGraw-Hill, 1991.
[13] S. Xing and B.-P. Paris, "Measuring the size of the Internet via importance sampling—biasing through thresholding," in *Proc. 36th Conf. Information Sciences Systems*, Princeton, NJ, Mar. 2002, pp. 796–801.
[14] L. Kleinrock, *Queueing Systems, Volume I: Theory*. New York: Wiley, 1975.



Song Xing received the B.S. and M.S. degrees in electrical engineering from Southeast University, China, in 1985 and 1990, respectively. He is currently working toward the Ph.D. degree in electrical and computer engineering at George Mason University, Fairfax, VA.

From 1985 to 1995, he was a Lecturer in the Radio Engineering Department, Southeast University and also a Researcher at the National Mobile Communications Research Laboratory, China, from 1990 to 1995. He was a Visiting Researcher at the Signal Processing and Interpretation (SPI) Laboratory in the Electrical and Computer Engineering Department, Boston University, Boston, MA, from 1995 to 1996. He has accepted an Assistant Professor appointment in the Information Systems Department at California State University, Los Angeles. His research interests include Internet traffic and performance measurement, importance sampling simulations of stochastic systems, speech processing, and communication systems.

Bernd-Peter Paris received the Diplom-Ingenieur degree in electrical engineering from Ruhr-University Bochum, Germany, in 1986 and the Ph.D. degree in electrical and computer engineering from Rice University, Houston, TX, in 1990.

After being with the Public Switching Division at Siemens, Munich, Germany, for one year, he accepted a faculty appointment at George Mason University, Fairfax, VA, where he is currently an Associate Professor in electrical and computer engineering. He is engaged in research in the area of communication systems, with emphasis on communications networks, mobile, wireless communications, and information theory.

Dr. Paris is the recipient of a Fulbright Scholarship in 1986 and of the National Science Foundation (NSF) Research Initiation Award (1993–1996). He is a Member of Eta Kappa Nu.