Available online at www.sciencedirect.com
SCIENCE DIRECT
Computer Communications 28 (2005) 880–897
computer communications
www.elsevier.com/locate/comcom

An investigation of web crawler behavior: characterization and metrics

Marios D. Dikaiakos[a,*], Athena Stassopoulou[b], Loizos Papageorgiou[a]

[a]Department of Computer Science, University of Cyprus, P.O. Box 20537, Kallipoleos 75, Nicosia 1678, Cyprus
[b]Department of Computer Science, Intercollege, P.O. Box 24005, Nicosia, Cyprus

Received 22 July 2004; revised 9 January 2005; accepted 11 January 2005
Available online 25 February 2005

Abstract

In this paper, we present a characterization study of search-engine crawlers. For the purposes of our work, we use Web-server access logs from five academic sites in three different countries. Based on these logs, we analyze the activity of different crawlers that belong to five search engines: Google, AltaVista, Inktomi, FastSearch and CiteSeer. We compare crawler behavior to the characteristics of the general World-Wide Web traffic and to general characterization studies. We analyze crawler requests to derive insights into the behavior and strategy of crawlers. We propose a set of simple metrics that describe qualitative characteristics of crawler behavior, vis-à-vis a crawler's preference on resources of a particular format, its frequency of visits on a Web site, and the pervasiveness of its visits to a particular site. To the best of our knowledge, this is the first extensive and in depth characterization of search-engine crawlers. Our results and observations provide useful insights into crawler behavior and serve as basis of our ongoing work on the automatic detection of Web crawlers.

Keywords: Web characterization; Crawlers

1. Introduction

A *Web crawler* is a program that traverses the hypertext structure of the Web, starting from a 'seed' list of hyper-documents and recursively retrieving documents accessible from that list [7,13,29]. Web crawlers are also referred to as robots, wanderers, or spiders. All major search engines employ powerful crawlers that traverse the Internet continuously, trying to discover and retrieve as many Web pages as possible. More recently, crawler systems have been used also as tools for focused crawling, for shopbot implementation, and for supporting added-value services on the Web (portals, personalized and mobile services, etc.) [14,19,29]. As a consequence, the number and the variety of active robots operating on the Internet increases continuously, resulting to a noticeable impact on WWW traffic and Web-server activity.

In this paper, we seek to characterize the behavior of crawlers that belong to popular Web-search engines. In particular, we investigate statistically various properties of crawler-induced HTTP traffic, such as the distribution of HTTP requests and reply-codes, the type and size of resources sought and retrieved, the distribution of crawler requests across a Web site, the frequency and pattern of crawler re-visits, and the temporal characteristics of crawler activity. We use these analyses to discover, characterize and compare the capabilities of different crawlers regarding their usage of caching, their capability to detect and avoid broken or dead links, their success in discovering Web-site resources, etc. Furthermore, we propose simple metrics that can be used to describe concisely aspects of crawling strategy and to compare basic crawler characteristics. Characterizing crawler activity is important as it enables researchers to: (i) estimate the impact of robots on the workload and performance of Web servers; (ii) investigate the contribution of crawlers to WWW traffic; (iii) discover and compare the strategies employed by different crawlers to reap resources from the Web, and (iv) model the activity of robots to produce synthetic crawler workloads for simulation studies. Finally, characterization can be the basis for the automatic detection of robots.

For the purposes of our study, we focus on the characterization of five crawlers that belong to well-known

* Corresponding author. Tel.: +357 22892700; fax: +357 22892701.
E-mail addresses: mdd@ucy.ac.cy (M.D. Dikaiakos), stassopoulou.a@intercollege.ac.cy (A. Stassopoulou), loipap@ucy.ac.cy (L. Papageorgiou).

0140-3664/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2005.01.003

search engines and digital libraries. Four of them belong to major, general-purpose engines: *Google* [4,13], *AltaVista* [2], *Inktomi* [5], and *FastSearch* [3]. The fifth crawler belongs to *CiteSeer*, also known as 'ResearchIndex,' the Digital Library and Citation Index of NEC Research Institute and Penn State University [1]. To study the behavior of these crawlers, we employ and analyze the access logs of five Web servers hosting academic and research sites in three different countries.

The remaining of this paper is organized as follows: Section 2 gives an overview of the logs used in our study and introduces ALAN, a system that we developed to capture crawler activity and study Web logs. In Section 3, we present the characterization of crawler activity. Proposed metrics are discussed in Section 4. In Section 5, we discuss related work. Finally, Section 5 presents a summary of our observations and conclusions.

2. Access log analysis

To characterize statistically the behavior of a Web crawler, we need to examine the HTTP interactions that take place between the crawler and a set of Web servers during a prolonged period of time. Typically, these interactions are recorded in Web-server access logs. For the purposes of our study, we use access logs from:

- The University of Cyprus; one set from the departmental Web server of the Computer Science Department (log acronym *CS-UCY*) and one set from the main University Web server (*CC-UCY*). The CS-UCY log captures the traffic of 176 days and contains requests for 69,918 documents. The CC-UCY log captures the traffic of 114 days and contains requests for 47,751 distinct documents.
- The Institute of Computer Science, Foundation of Research and Technology, Hellas in Greece (*ICS-FORTH*); this log corresponds to a period of 45 days and captures requests for 58,225 distinct documents.
- The Software Engineering Laboratory server at the National Technical University of Athens, Greece (*SL-NTUA*); the SL-NTUA log corresponds to a period of 58 days and captures requests for 102,088 distinct documents.
- The departmental server of Computer Science and Engineering at the University of Toronto, Canada (*CSE-TOR*); this log corresponds to a period of 42 days and captures HTTP requests for 48,229 distinct documents.

These logs capture the HTTP traffic of a period spanning from the fall of 2001 to the winter of 2002; log durations range from 42 to 176 days. Overall, our logs contain a total of 9.3 million HTTP requests for 326,211 distinct URL's, resulting to a total of 812 GB of transferred data. A detailed overview of the log-suite employed is given in Table 1.

To proceed with our analysis, we need to pre-process our logs in order to identify and extract access-log entries corresponding to HTTP transactions initiated by IP addresses that belong to the five crawlers under investigation. Consequently, publicly available, anonymized logs, which hide the originating IP addresses of recorded HTTP requests, and which are typically used in Web characterization studies [17,26], are of no particular use for this study. Therefore, we have to rely on the non-publicly available access logs mentioned above, which were given to us under a non-disclosure agreement protecting the privacy of end-users that accessed the respective sites. It should also be noted that these logs belong to academic and research institutions; unfortunately, it was not possible to get access logs from ISPs or commercial portals, since private institutions consider such logs as sensitive information.

To process Web-server access-logs, we designed and developed **ALAN** (A Log ANalyzer). ALAN is a library written in JAVA, which provides classes and methods for pre-processing and filtering access-logs. ALAN produces output compatible to Matlab. Access-log processing through ALAN comprises two parts: pre-processing and result extraction. Pre-processing aims to clean the records, convert them to a more convenient format, and provide some useful

Table 1
Summary of access log characteristics

Log acronym	CS-UCY	CC-UCY	ICS-FORTH	SL-NTUA	CSE-TOR
Log origin	CS, UCY	CC, UCY	ICS, FORTH	SOFTLAB, NTUA	CSE, U. of Toronto
Country code	CY	CY	GR	GR	CA
Log duration (days)	176	114	45	58	42
Starting date	11/9/01	15/1/02	11/3/02	1/1/02	13/2/02
Ending date	6/3/02	9/3/02	25/4/02	27/2/02	27/3/02
Log size (MB)	184.8	150.7	81.8	525.9	243.7
Total requests	1,767,101	1,467,266	786,300	2,724,074	2,565,214
Distinct URL's requested	69,918	47,751	58,225	102,088	48,229
Avg requests/day	10,040.35	12,850.46	17,473.33	46,966.8	61,076.52
Bytes transferred (MB)	23,618.53	18,946.92	8021.12	745,387.42	36,234.55
Avg bytes/day (MB)	134.20	166.20	178.25	12,851.51	862.73

Table 2
Identification of known crawlers from IP addresses

Crawler name	Well-known substrings	Hostnames discovered	IP addresses extracted
Google	googlebot.com	12	125
Inktomi	inktomi.com	106	100
Alta vista	sv.av.com	34	31

indications for further processing. It comprises the following steps:

1. Removal of incomplete and erroneous records.
2. Identification of IP addresses that belong to crawlers of interest. To this end, we use ALAN to collect all unique originating IP addresses of HTTP requests recorded in the logs and perform reverse DNS lookups to convert IP addresses to hostnames. Using this mapping, we assign IP addresses to crawlers of interest, with a well known set of hostnames. For example, Google's crawlers use hostnames containing the sub-string '.googlebot.-.googlebot.com.' Then, we use ALAN to prepare a table with all the IP addresses that belong to each crawler. Table 2 indicates the quantitative results extracted from the five server log files considered. Notably, not all IP addresses specified in the log files were used by the crawlers at the same time. Therefore, certain hostnames may correspond to more than one IP addresses or an IP address may correspond to more than one hostnames.
3. Scanning the log files and collecting all crawler requests. The outcome of this process is a log file containing the requests that emanate from the crawler at hand.

Following the pre-processing step, we invoke the analysis phase of ALAN, which produces statistical data for inter-arrival times, transferred bytes, HTTP methods, and response codes. Distributions of inter-arrival times and transferred bytes are visualized in order to discover a possible match with well known statistical distributions (before proceeding to statistical tests). Furthermore, the temporal activity of crawler IPs is visualized in order to reveal possible time-depended properties, like periodic behavior.

ALAN is implemented in Java. It employs multithreading and serial manipulation of files (especially before extracting the requests of interest). The DNS lookup process is multithreaded and synchronized—up to 500 threads can be used—in order to cope with the latency that each DNS lookup request causes. ALAN methods produce output compatible to Matlab, which was used for statistical tests and visualization of the results.

Running ALAN on a 1.5 GHz/256 MB PC in Windows 2000 environment resulted to a pre-processing speed of around 28,000 records per second. Further analysis of inter-arrival times and transferred bytes was done at a throughput of 250,000 records per second.

3. Crawler characterization

To characterize the behavior of the five Web crawlers of our study, we examine:

1. The characteristics of crawler-induced HTTP traffic; for instance, the distribution of HTTP-request and reply codes carried by the HTTP messages exchanged between crawlers and Web servers.
2. Features of the Web resources discovered and retrieved, such as their format, size, and percentage over other types of accessible resources.
3. Temporal properties that unveil the timing of crawler requests. For example, the arrival rate and the periodicity of crawler requests, and the distribution of inter-arrival times.

Furthermore, we compare these characteristics against the corresponding traits of the general HTTP traffic, in order to identify features that distinguish between crawler-initiated and user-initiated HTTP requests.

We used ALAN to extract the activity of the five crawlers from our access-log suite. We give an overview of this activity in Table 3. If we take into account all crawlers and logs, we end up with a total of 792,285 crawler-induced requests that generate a traffic of 5 GB of data; individual crawlers generate HTTP traffic of 1.1 to 33.52 MB/day on each Web server examined. Collectively, the activity of the five crawlers represents the 8.51% of the total number of requests included in our logs and the 0.65% of the bytes transferred. The impact of the five crawlers on each individual Web server is presented in Table 4. From this table we can see that in four out of the five sites, the five crawlers are responsible for the 9.48–12.67% of the total incoming requests and for the 4.33–5.84% of outgoing traffic, which represent a sizable proportion of the overall HTTP activity in these particular servers. Crawler contribution to the outgoing traffic in the fifth site (SL-NTUA) is negligible; this is because the SL-NTUA server hosts very large and very popular multimedia files, which are of no interest to the crawlers of our study.

3.1. HTTP messages

As a first look into Web crawler behavior, we examine the differences between *crawler-induced* and *general* HTTP traffic. A statistical analysis of the general HTTP traffic captured in our logs results in observations similar to those published in prior studies [9,8,22]; our findings are summarized in Table 5 and Fig. 1.

We observe that nearly all crawler-induced HTTP messages carry GET requests. Also, we observe that crawlers that implement caching, such as Google, Inktomi, and FastSearch, issue cache-validation commands at a rate much higher than the rate observed in the general population of WWW clients: 42.21, 33.14 and 33.17% versus 16.47%.

Table 3
Activity overview of selected crawlers (Google, Inktomi, Altavista, FastSearch, CiteSeer)

Log acronym	CS-UCY	CC-UCY	ICS-FORTH	SL-NTUA	CSE-TOR
Google					
Total requests	28,082	25,949	9269	11,147	8354
Distinct requests	7350	5937	6206	7614	6327
Total bytes transferred (MB)	832.06	340.73	179.94	78.01	315.44
Avg requests/day	159.6	227.56	205.98	193.3	199.28
Avg bytes/day (MB)	4.73	2.99	3.4	1.35	7.51
Inktomi					
Total requests	44,372	54,263	51,030	30,080	24,097
Distinct requests	7089	4637	35,097	17,636	11,823
Total bytes transferred (MB)	186.63	175.43	49.97	203.11	65.67
Avg requests/day	252.27	475,99	1134	521.68	574.97
Avg bytes/day (MB)	1.06	1.54	1.11	3.50	1.56
AltaVista					
Total requests	101,809	30,721	37,355	39,698	21,769
Distinct requests	31,452	7390	17,004	22,751	9497
Total bytes transferred (MB)	299.92	277.26	103.83	199.85	1407.99
Avg requests/day	578.8	269,48	830.11	688.5	5194.23
Avg bytes/day (MB)	1.70	2.4	2.31	3.45	33.52
FastSearch					
Total requests	7575	27,935	1829	28,533	10,712
Distinct requests	5642	6660	1566	11,743	8826
Total bytes transferred (MB)	12.36	171.86	6.84	179.64	302.17
Avg requests/day	43.04	245.04	40.64	491.95	255.05
Avg bytes/day (MB)	0.07	1.51	0.15	3.10	7.19
CiteSeer					
Total requests	675	361	135	64	141
Distinct requests	669	361	135	64	135
Total bytes transferred (MB)	21.34	2.64	6.82	1.07	15.95
Avg requests/day	3.84	3.17	3	11	3.36
Avg bytes/day (MB)	0.12	0.02	0.15	0.02	0.38

Furthermore, we observe that responses to crawler requests exhibit a proportion of 4xx error codes higher than the observed rate for all clients. Most of the error codes are due to unavailable resources '404 Not found.' The higher rate of 4xx codes can be explained by the fact that human users are able to recognize, memorize and avoid erroneous links, unavailable resources, temporarily unavailable servers, etc. It is the (rational) behavior and choices of those users that determine the all-clients characterization.

3.2. Retrieved resources

3.2.1. Resource-type distribution

For most Web sites, the resources that receive the overwhelming majority of requests are text (text/plain, text/html) and image files (image/jpeg, image/gif, etc.) [8,22]. The remaining content types constitute a relatively small portion of requested URL resources (postscript and PDF, audio and video, scripts, applets). As we can see from Fig. 2 ('All Clients' bars), over 90% of all requests in four out of the five logs of our test-suite target text or image resources.

The situation is different if we focus on requests arising from the five crawlers at hand. As we can see from Fig. 2, text-file requests represent the 71.67–97.22% of total requests, whereas requests for image resources are practically non-existent. Finally, crawlers such as Google and NEC's CiteSeer, which index textual and non-textual documents (e.g. postscript, pdf, compressed files) pursue the retrieval of the corresponding URL resources much

Table 4
Contribution of selected crawlers to Web-server activity

Log acronym	CS-UCY (%)	CC-UCY (%)	ICS-FORTH (%)	SL-NTUA (%)	CSE-TOR (%)
% of total requests	10.32	9.48	12.67	4.02	10.18
% of bytes	5.72	5.11	4.33	0.08	5.84

Table 5
Percentage of HTTP responses to selected crawlers over all logs

Response codes	2xx (%)	304 (%)	3xx (except 304) (%)	4xx (%)
All dients	72.26	16.47	1.98	9.29
Google	41.86	42.21	3.31	16
Inktomi	33.73	33.14	7.4	25.7
AltaVista	80.18	0	0.53	19.28
Fastsearch	52.58	33.17	2.25	11.99
CiteSeer	59.59	1.09	4.21	35.10

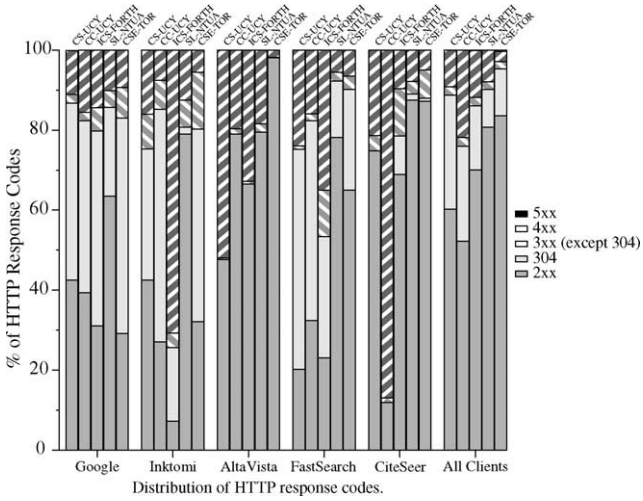

Fig. 1. Distribution of HTTP response codes.

more aggressively than the general population of Web clients.

3.2.2. Resource-size distribution

Prior Web characterization studies have shown that the size of resources transported through the HTTP protocol is on the average relatively small but highly variable. Our analysis revealed similar observations, with the exception of the SL-NTUA access-log (see Table 6), whose Web server turned out to serve very large multimedia documents (for more information see [18]). The distribution of resource sizes and HTTP-response messages is represented by a hybrid model that describes the body of the distribution with a *log-normal* and the tail with a *heavy tailed* (*Pareto*) distribution [8,10,15,22].

If we concentrate on crawler traffic only, we find that HTTP responses to crawler requests have an average size of 7.03 KB; as we can see from Table 6, however, there is a high variability across different crawlers and access-logs.

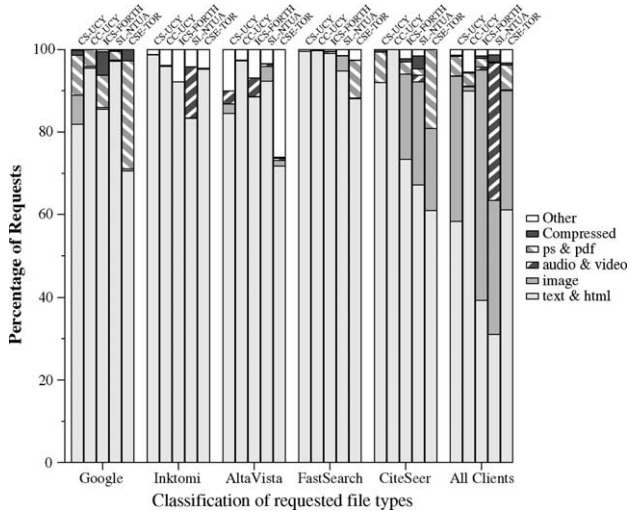

Fig. 2. Classification of requested resource types.

Table 6
Average size of HTTP responses (in KB)

Log acronym	CS-UCY	CC-UCY	ICS-FORTH	SL-NTUA	CSE-TOR
All clients	13.69	13.22	10.45	280.20	14.46
Google	30.34	13.45	19.83	7.17	38.67
Inktomi	4.31	3.31	1	6.91	3.12
AltaVista	3.02	9.24	2.85	5.12	6.62
FastSearch	1.67	6.30	3.83	6.45	28.89
CiteSeer	32.37	7.49	51.75	17.12	115.82

This is attributed to the fact that, in contrast to Inktomi and AltaVista, Google and CiteSeer download postscript, pdf, and image resources, which have larger average sizes (see also Fig. 2). Further evidence for the response-size variability to crawler requests can be derived if we compare the mean to the median values of HTTP responses. For instance, looking at the ICS-FORTH logs, the mean transfer size of responses that carry HTML resources is 5.53 KB, whereas the median is 0.19 KB. Similar observations hold for other types of URL resources and other logs.

A significant portion of the HTTP traffic corresponds to messages carrying no content and having a very small size. For instance, over 40% of Google messages have 3xx and 4xx codes. Therefore, it is interesting to study the size distribution of successful messages with a 200 OK code; this will provide insights on the size and type of content *downloaded* by users and crawlers. Figs. 3–5 present diagrams with the body and the tail distribution of the sizes of successful responses to all five crawlers, which provide evidence for a high variability in the sizes of resources retrieved by Web crawlers. The size of downloaded resources can be described by a hybrid log-normal and Pareto distribution.

3.2.3. Distinct requests

When studying the patterns of URL requests that arrive at a particular Web server, it is interesting to estimate the percentage of *separate* (distinct) resources requested over the *total* number of requested resources, and the percentage of resources requested only once (the 'one-timers') [8,9]. These percentages provide insights on the benefits of Web caching strategies, and a characterization of the 'repetitive' nature of requests arriving from the Web. Very small percentages of distinct URL-requests imply that there is potential for performance improvement through caching of Web documents on the Web server or within the network. The maximum possible effectiveness of caching is limited, however, by the percentage of 'one-timers.'

Table 7 presents the percentage of distinct requests for the logs under investigation. When we take into account requests from all clients, this percentage is small and ranges between 1.88 and 7.4%. This observation agrees with prior Web characterization studies (e.g. [8,9]). Nevertheless, it changes drastically if we focus on requests arriving from IP addresses that belong to

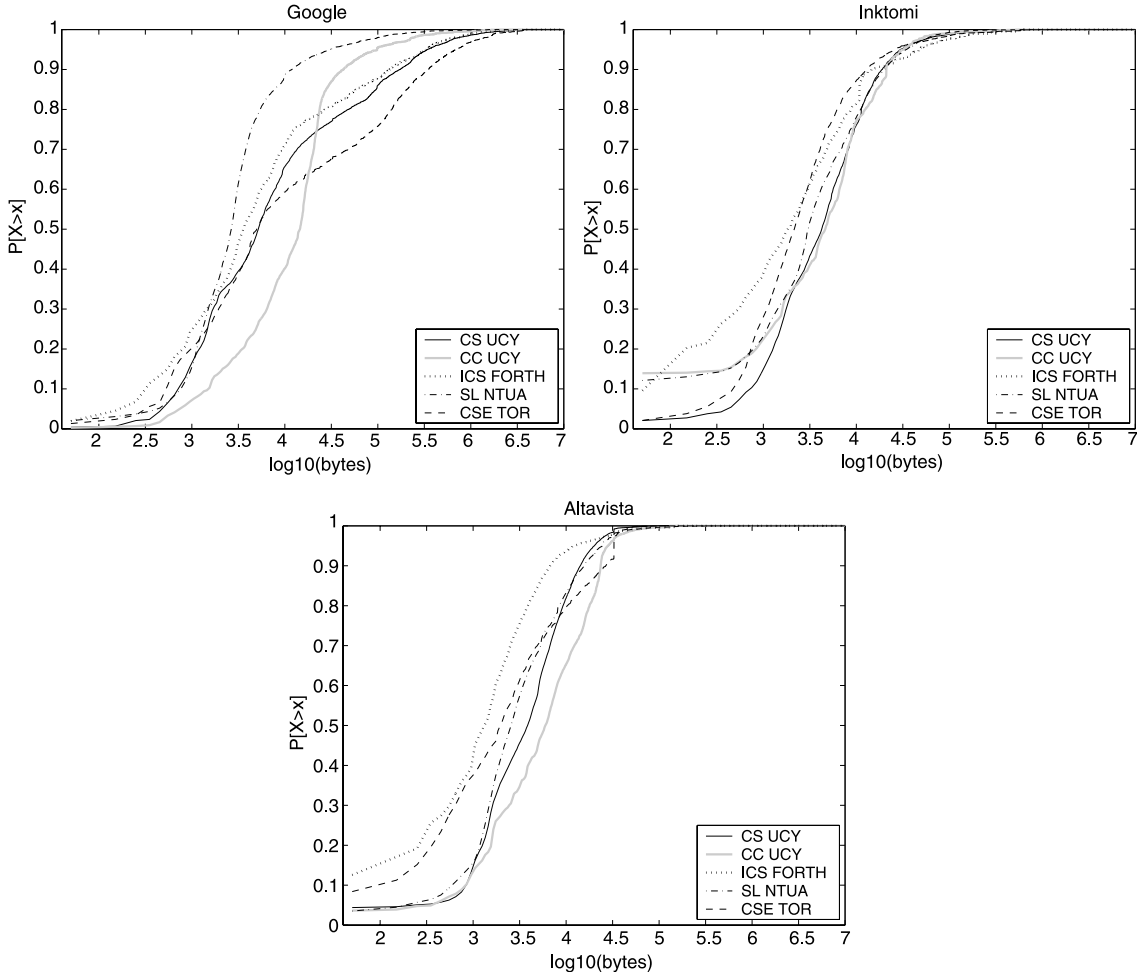

Fig. 3. Size distribution for successful responses: cumulative histograms (Google, Inktomi, AltaVista).

individual crawlers: percentages increase by an order of magnitude and up to 100%. This is because the percentage of distinct requests over the total requests coming from a crawler depends on the (typically limited) number of visits this particular crawler pays to the Web site at hand, within the time frame captured by the access log under study. For instance, in the period captured by the ICS-FORTH log, CiteSeer visits ICS's

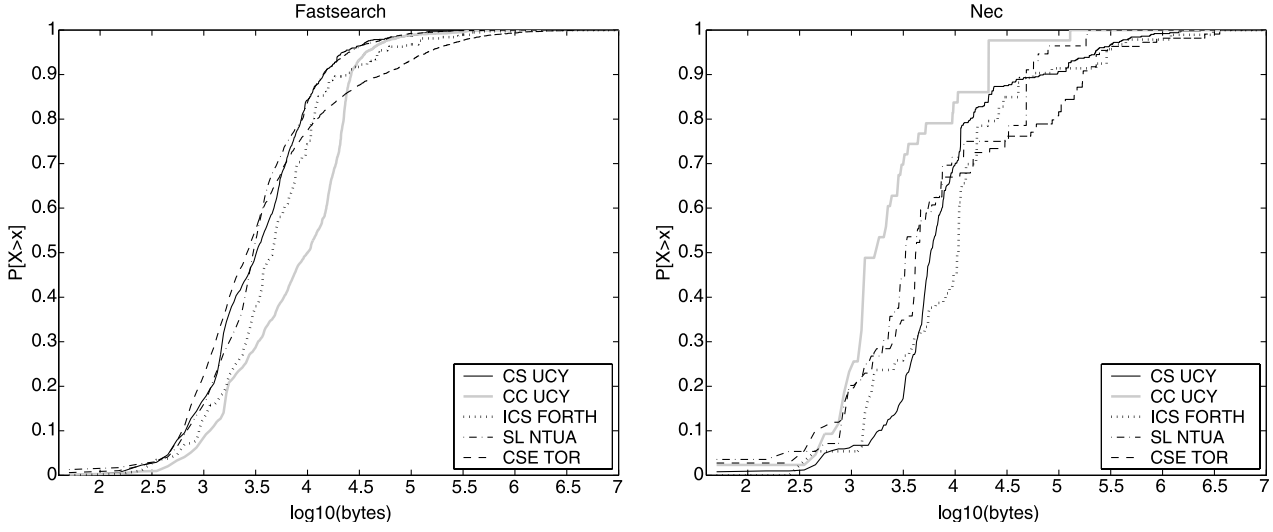

Fig. 4. Size distribution for successful responses: cumulative histograms (FastSearch, CiteSeer).

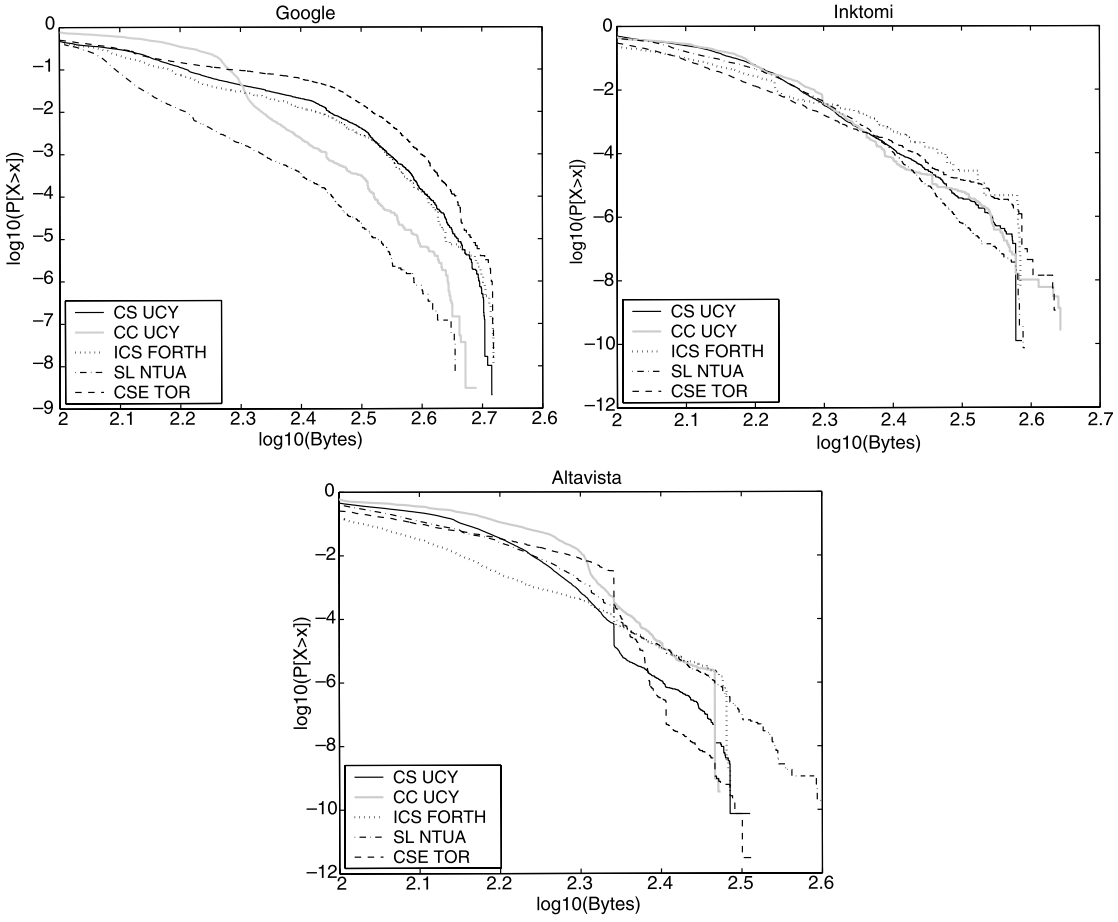

Fig. 5. Size distribution for successful responses: heavy tails (Google, Inktomi, AltaVista).

Web site only once and the corresponding percentage is 100% (see Table 7).

3.2.4. Popularity of resources referenced by crawlers

In prior Web characterization studies, the *popularity* of a URL resource is measured as the number of requests accessing that resource over the total number of requests reaching its Web site. These studies have shown that if we rank resources of a Web site in decreasing order of popularity, the proportion of requests for a resource is inversely proportional to its rank. Hence, resource-popularity follows a *Zipf*-like distribution [8,11,12,22]. To test if a distribution is Zipf-like, we produce a log–log plot of the number of requests for each resource versus the resource's popularity rank. Resources are placed on the horizontal axis in decreasing order of rank; if the distribution is Zipf-like, the graph should appear linear with a slope of -0.5 to -1 [22,8].

Figs. 6 and 7 present popularity plots for the distinct resources in our logs versus the rank of these resources. The left diagram of Fig. 6 presents the number of requests from *all clients* versus the resources sorted in decreasing order of their rank, in log–log scale. From this graph we can see that the plots can be segmented in roughly three different areas. The first area includes flat regions of very popular resources exhibiting nearly equal popularity. The second area is Zipf-like; the popularity plot can be fitted to a linear diagram with negative slope near -1. The last area includes resources with very small popularity, which drops with a slope much smaller than -1.

We calculate the 'popularity' of Web resources based on *crawler requests only* and plot our findings in the diagrams of Figs. 6 and 7. Notably, these popularity figures provide a measure of how the repeated crawler requests spread across available resources. This measure has nothing to do with the actual popularity that search-engine users show for these resources. Statistical observations for the case of

Table 7
Percentage of requests for distinct URL resources over total requests

Log acronym	CS-UCY (%)	CC-UCY (%)	ICS-FORTH (%)	SL-NTUA (%)	CSE-TOR (%)
All clients	3.96	3.25	7.4	3.75	1.88
Google	26.17	22.89	66.95	68.31	75.74
Inktomi	15.98	8.55	68.78	58.63	48.97
AltaVista	30.89	24.06	45.52	57.31	43.63
FastSearch	74.48	23.84	85.62	41.16	82.39
CiteSeer	99.11	100	100	100	95.74

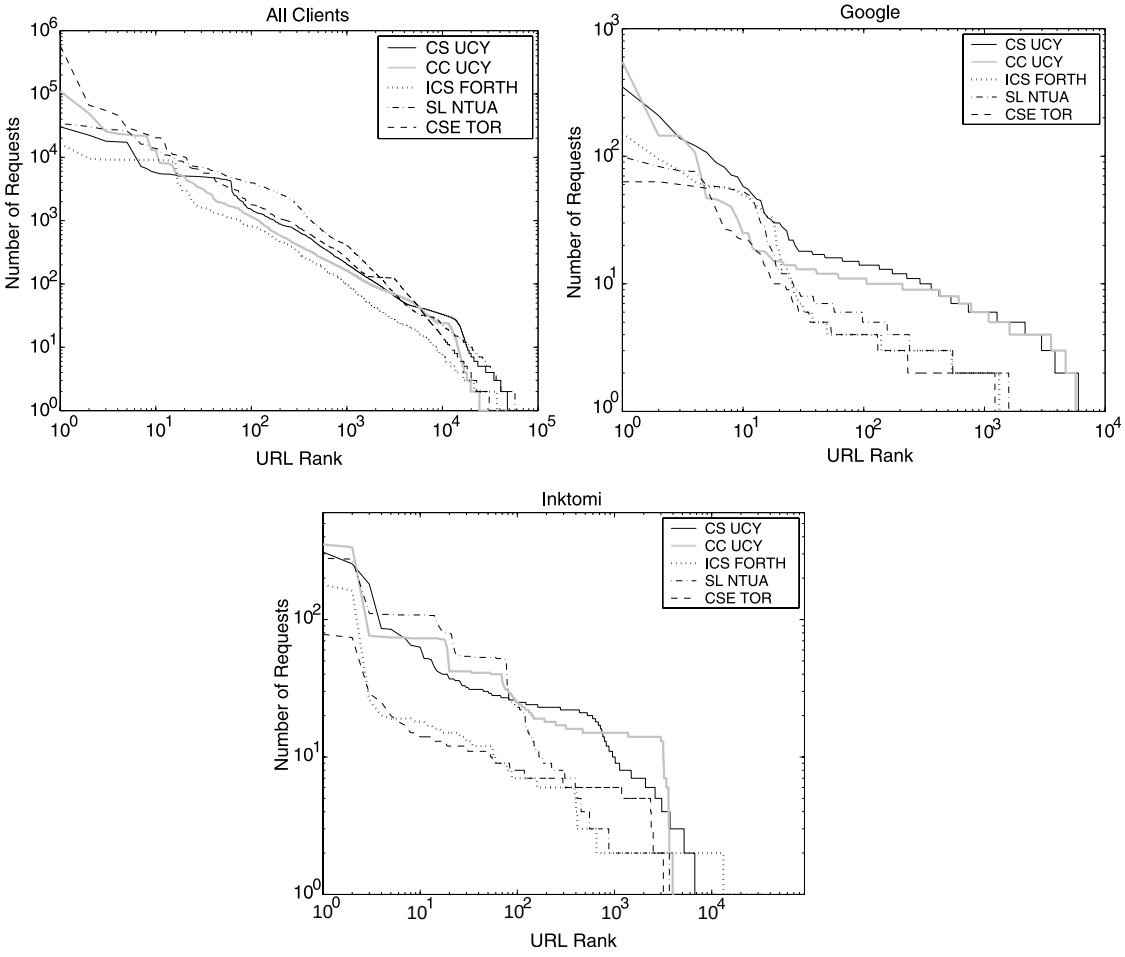

Fig. 6. Resource popularity (All clients, Google, Inktomi).

crawler-logs are harder as the total number of requests issued by each crawler during our period of observation is small, i.e. one to two orders of magnitude smaller than the total number of requests coming from all clients. These crawler plots, however, provide insights into crawler referencing patterns. Notably, many of the popularity diagrams display a step-wise shape with URL resources clustered into smaller subsets of equal popularity. In other words, the frequency of visits of a crawler on a particular Web site varies for different subsets of resources within the site.

3.2.5. Concentration of requests

Popularity studies for WWW access show that URL requests are highly concentrated around a small set of resources. The *concentration* of requests can be expressed by sorting the list of distinct URL resources requested into decreasing order of rank, and then plotting the cumulative frequency of requests versus the *fraction* of the total URL resources requested [9]. Previous Web characterization studies have shown that resource-popularity is highly concentrated [8,11,12,22]. We conduct the same analysis here and present our results in Figs. 8 and 9. The left diagram of Fig. 8 presents the cumulative distribution of requests versus the percentage of distinct resources for all clients; resources are taken in decreasing order of rank. As we can easily see from this diagram, our logs exhibit a high concentration of references on a small subset of unique resources: 10% of separate (distinct) URL resources attract a 75–90% of all requests. It is interesting to note that servers with a smaller daily rate of incoming HTTP requests, such as CS-UCY and CC-UCY, depict a smaller concentration of references if compared to more busy servers like CSE-TOR and SL-NTUA.

Focusing on crawler-induced requests, however, it becomes apparent that crawler-references are *not* highly concentrated around a small set of URL resources. For instance, in the vast majority of cases, 50% of the most popular resources attract between 60 and 80% of all crawler-induced requests (see the two rightmost diagrams of Fig. 8 and diagrams of Fig. 9). This behavior is expected in the case of crawlers that try to reach as many resources as possible when visiting a particular Web site, without making any distinction between different resources.

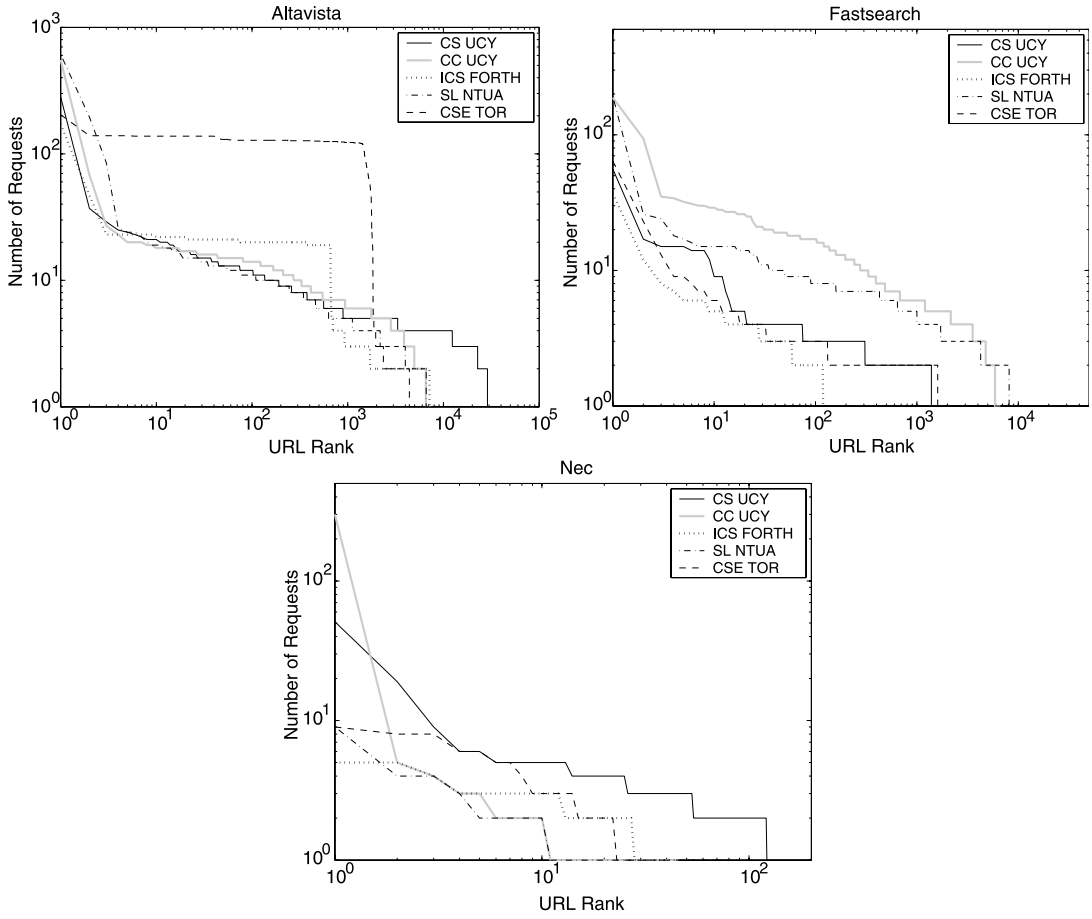

Fig. 7. Resource popularity (AltaVista, FastSearch, NEC's CiteSeer).

3.3. Temporal behavior

3.3.1. Distribution of inter-arrival times

One of the most important aspects of Web workload characterization lies in the temporal patterns of Web-request arrivals. Earlier studies have shown that TCP-connection arrivals that are heavily influenced by the HTTP traffic, can be modeled by the heavy-tailed Weibull distribution [20]. Also, that HTTP traffic is bursty and highly variable, and that inter-arrival times of HTTP requests are heavy tailed [15,22].

To investigate the inter-arrival-time distribution of crawler requests, we process our logs to measure and extract all time-intervals between successive HTTP requests issued by a particular crawler. As noted earlier, a crawler employs multiple fetchers to crawl a site and different fetchers may reside on different IP addresses. Therefore, we take into consideration requests coming from all IP addresses identified with that crawler and study the statistical characteristics of the union of all corresponding time-intervals.

In Figs. 10–12, we present logarithmic diagrams of the empirical density of inter-arrival times of HTTP requests from Google, AltaVista and Inktomi on CS-UCY, ICS-FORTH and CSE-TOR. From these diagrams we can see that the time between subsequent HTTP requests is highly non-uniform and heavy-tailed. The observed distributions reflect the presence of multiple underlying distributions representing the behavior of fetcher-processes residing at different IP hosts of the same crawler. This effect is more pronounced for Google and Inktomi, which use a very high number of different IP hosts, than for AltaVista. Furthermore, the inter-arrival-time distribution of requests coming from an individual IP host is the combination of two underlying distributions: the first distribution represents the inter-arrival times of HTTP requests generated by the fetcher-process(es) of this IP host within one 'crawling-session.' The second represents the times between subsequent crawling-sessions. Shorter inter-arrival times are observed within a crawling-session whereas longer intervals correspond to periods of 'silence,' or crawler inactivity.

3.3.2. Crawler periodicity

An interesting point that arises when investigating a crawler's activity is whether it exhibits a periodic behavior. By plotting the time activity (i.e. the active and inactive periods of time) of crawler processes issuing requests, we

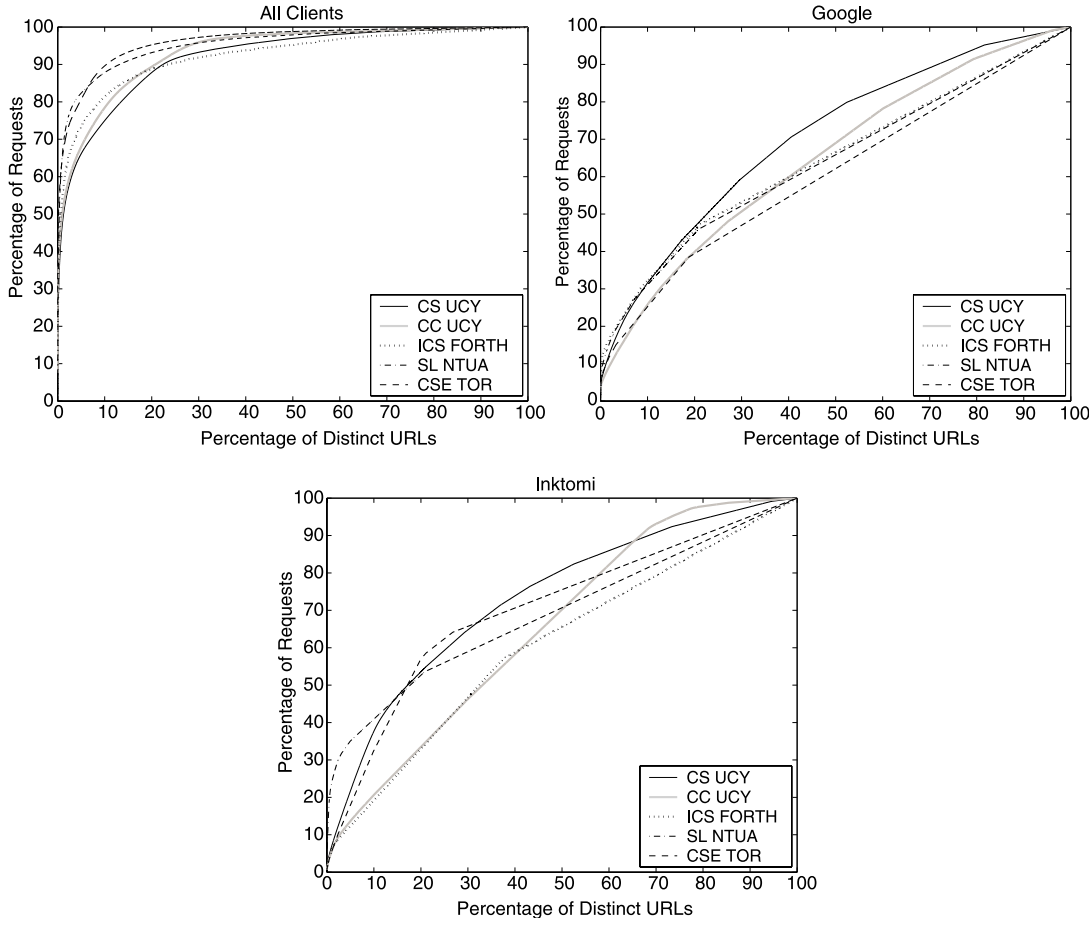

Fig. 8. Concentration of requests (All clients, Google, Inktomi).

observed that several of them seem to exhibit, at least partially, a periodic pattern. We investigated further this observation and verified the periodicity for several IP addresses belonging to crawlers and estimated their time cycles.

For this task, we used the Fast Fourier Transform (FFT). The FFT maps a function in the time field to a, complex in general, function in the frequency field. The idea is that by observing peaks of magnitude in the frequency field we can easily conclude that time activity has periodicity. The frequency coordinate of each possible peak is inversely proportional to the time cycle of the periodicity. Since we are not interested in the phase of the frequency plot, we illustrate the *spectral density function*, which is the square of the magnitude of the FFT.

Before implementing the FFT, we pre-process the requests issued from a certain IP address belonging to a crawler. Time is assumed to be sliced; we use a 10 s time interval (granularity). Ideally, the granularity should be as small as possible, but we tried to keep the number of resulting points relatively small for a faster FFT computation.

We count the requests issued from the IP address of interest in each time interval. Because our focus of interest at this stage is on the presence of some periodic action, we assign the value of one to the intervals that have at least one hit and the value of zero to the ones with zero hits. Consequently, we produce an ON-OFF signal that represents the crawler's time activity for the selected granularity. This signal is passed as input to the FFT function. The resulting diagrams reveal a periodicity in the requests issued by IP addresses belonging to several crawlers; in some cases this phenomenon is rather intense.

We present two examples: the first one analyzes the activity of an IP belonging to AltaVista and hitting CSE-TOR; the second analyzes the activity of a Google IP hitting CS-UCY. Fig. 13 (left) presents the ON-OFF signal of an AltaVista IP address at CSE-TOR and indicates that there is periodicity. In Fig. 13 (right), we plot the power spectral density function with respect to the inverse frequency (time). FFT specifies the main periods observed on that signal. For example, in Fig. 13 (left), from 7×10^5 up to 1.5×10^5 s we observe a periodic behavior which corresponds to the peak of around 8400 s in the right diagram of Fig. 13. Similar results are illustrated in Fig. 14, which presents the ON-OFF signal and the power spectral density, respectively for the Google IP address at CS-UCY. We observe a dominating period of about 2.5×10^6 s. We can

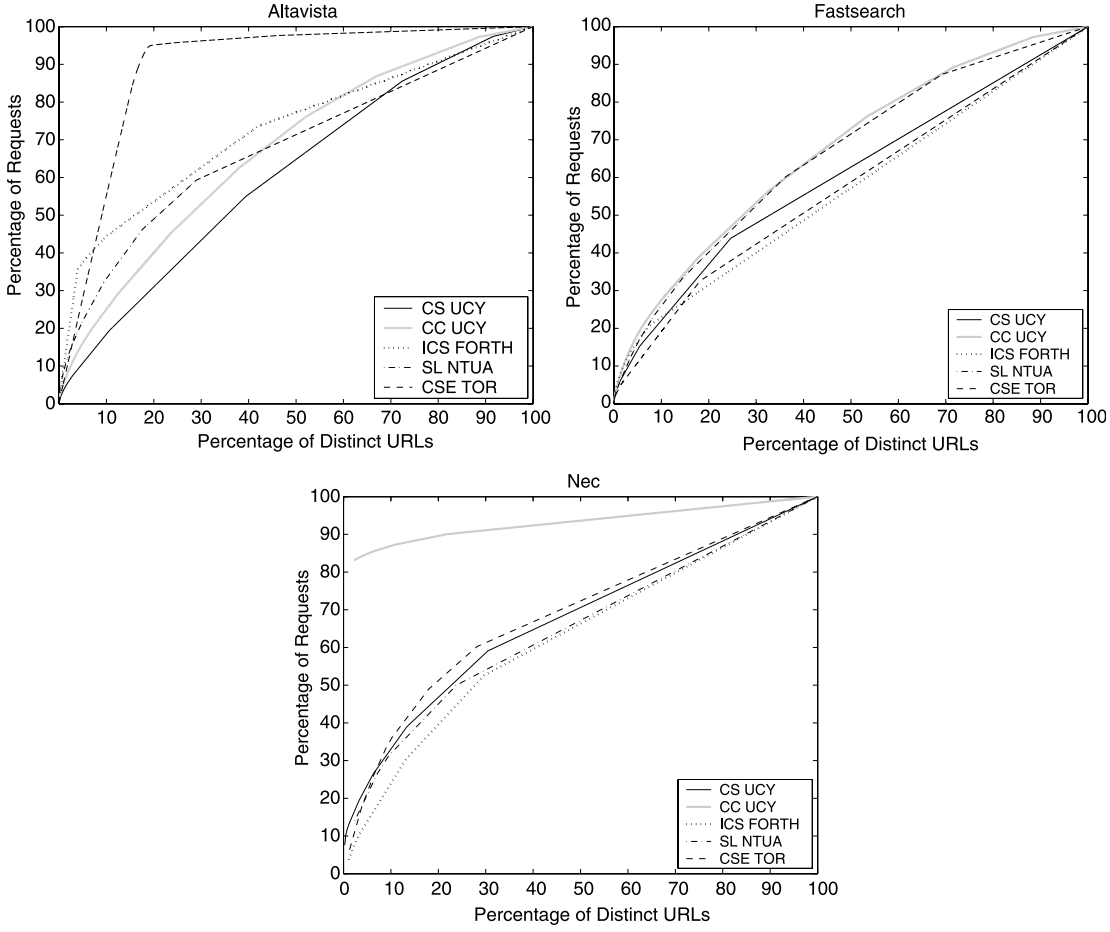

Fig. 9. Concentration of requests (AltaVista, FastSearch, NEC's CiteSeer).

therefore conclude that periodic activity can be expected from crawlers.

4. Metrics

In Section 3.3.2 we presented a characterization of crawler behavior, which enabled us to point out characteristics that are common across all crawlers and differences that exist between crawler-induced and general HTTP traffic. But what happens if we want to identify differences in the behavior of different crawlers? To this end, and based on earlier remarks, we propose the following metrics.

4.1. Format preference

To describe concisely the level of preference that a crawler shows toward resources of a particular format, we introduce the *format-preference metric*, m_{fp}. Format preference expresses the percentage of requests issued by a crawler for resources encoded in some format, normalized over the corresponding percentage for all clients visiting the same site at the same time-frame. This latter percentage represents an estimate of the 'average' stream of requests targeting resources of the format in question. The format-preference metric, m_{fp} is defined as follows:

Definition 1. (Format Preference metric) *The preference that a crawler $\mathcal{C}$ expresses for resources encoded in format $\mathcal{F}$ when visiting a Web site with access-log L, is defined as follows*

$$m_{fp}(\mathcal{F}, L, \mathcal{C}) = \begin{cases} \dfrac{N_{\mathcal{C},\mathcal{F}}(L)/N_{\mathcal{C}}(L)}{N_{\mathcal{F}}(L)/N(L)}, & \text{if } N_{\mathcal{F}}(L) \neq 0 (1) \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where

- $N_{\mathcal{C},\mathcal{F}}(L)/N_{\mathcal{C}}(L)$ *is the percentage of crawler-induced requests captured in access-log L that seek resources in format $\mathcal{F}$.*
- $N_{\mathcal{F}}(L)/N(L)$ *is the percentage of requests issued by all clients and captured in access-log L that seek resources in format $\mathcal{F}$.*

From this definition, we observe that a value of $m_{fp}(\mathcal{F}, L, \mathcal{C})$ much higher than one means that crawler $\mathcal{C}$ seeks resources of format $\mathcal{F}$ more aggressively than the general population of Web clients. A value of m_{fp} close to zero, on the other hand, means that either $\mathcal{C}$ does not download

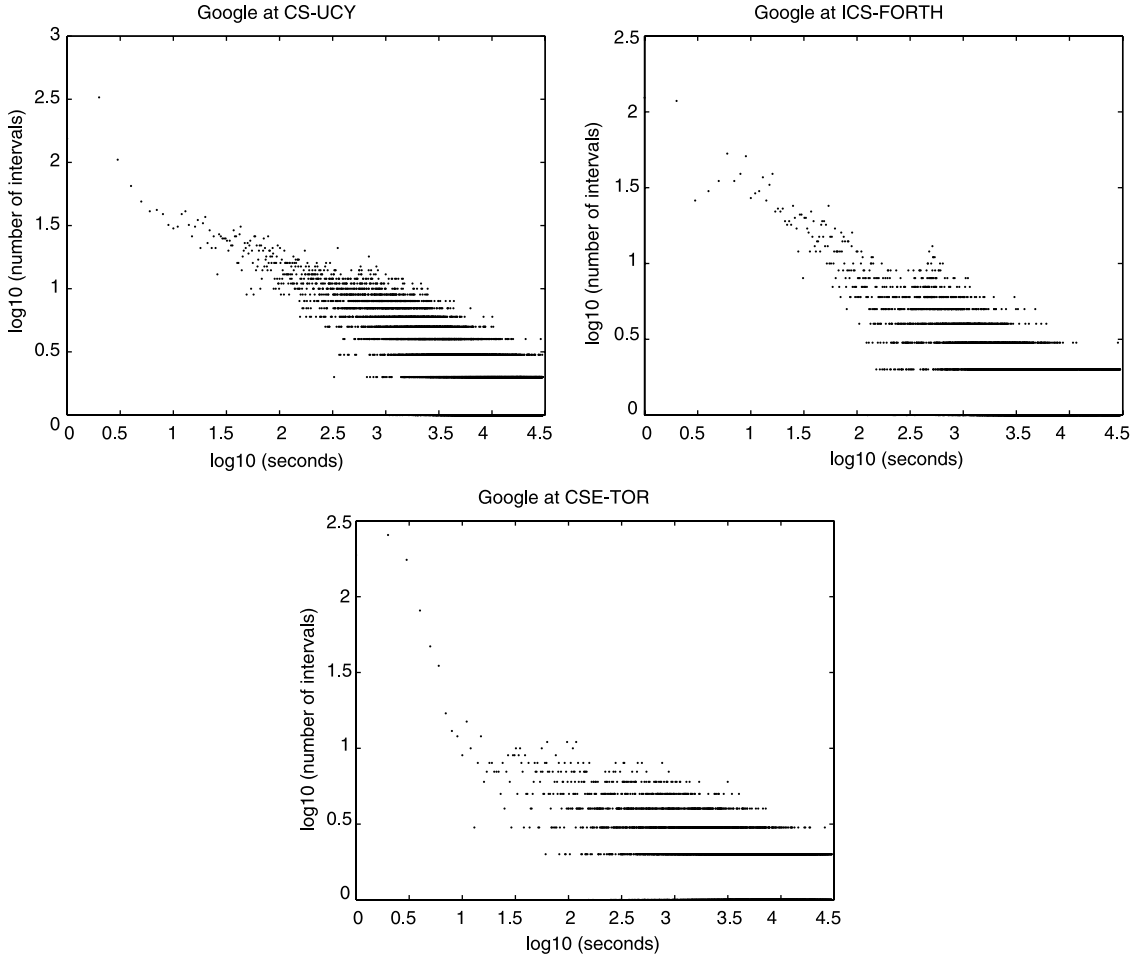

Fig. 10. Distribution of inter-arrival times for Google crawlers.

$\mathcal{F}$-resources, or that such resources are not available at the particular site. We should note, however, that the m_{fp} metric is very sensitive to the size of the log-sample.

The format preference metric can help us distinguish crawlers that try to collect audio or video files from those that collect HTML or PDF. For instance, in Fig. 15, we plot the values of the format preference metric for PS and PDF content for the five crawlers and logs of our study. From this diagram, we can easily observe that Google and CiteSeer collect postscript and PDF content, whereas other crawlers do not. This is in agreement with the functionality provided by the respective search engines.

4.2. Frequency of visits

As we observed earlier, the frequency of visits of a crawler on a particular site varies for different subgroups of that site's resources (see Figs. 6 and 7). It is interesting, however, to derive a rough estimate of how frequently a crawler repeats the crawl of a site within some time-frame. This estimate can be computed as the ratio of crawler requests for *distinct* URL resources over the total number of crawler requests. Such a measure, describes concisely how many times per month a crawler visits a site. Therefore, it can tell us if a crawler visits different sites with the same or different frequencies, and to compare the frequencies of visits of different crawlers. However, since logs from different sites capture different time periods, we need to normalize the frequency-of-visits metric by a time-normalization factor not smaller than the minimum time-frame of all the logs at hand. Here, we choose a normalization factor of 30 days (1 month). Consequently, the value of the *Frequency of Visits* metric will provide an approximation of the number of times a crawler visits a particular Web site within one month. The definition of the *Frequency of Visits metric*, m_{fov}, follows:

Definition 2. (Frequency of Visits metric) *The Frequency of Visit metric m_{fov} of a Web crawler $\mathcal{C}$, which visits a particular Web site with access-log L, is defined as follows*

$$m_{\text{fov}}(L, \mathcal{C}) = \frac{N_{\mathcal{C}}(L)/N^{d}_{\mathcal{C}}(L)}{t(L)} \times 30 \tag{2}$$

where

- *$N_{\mathcal{C}}(L)$ is the number of all requests issued by crawler $\mathcal{C}$ and recorded in access-log L;*

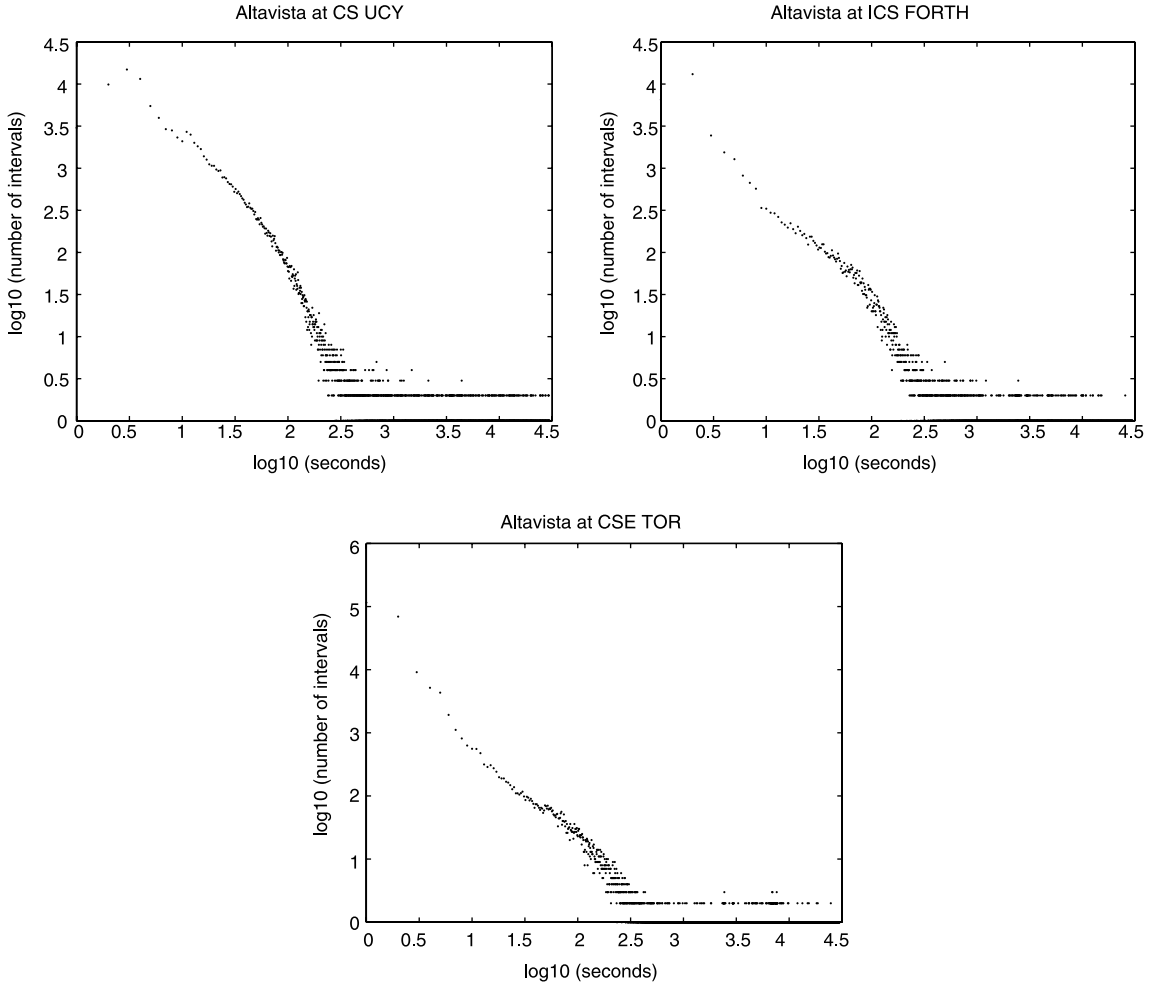

Fig. 11. Distribution of inter-arrival times for AltaVista crawlers.

- $N_{\mathcal{C}}^{d}(L)$ *is the number of requests for distinct URL resources issued by crawler* $\mathcal{C}$ *and recorded in access-log* L;
- $t(L)$ *is the time-period captured by access-log* L *(expressed in days)*;

Fig. 16 (left) presents a diagram of the Frequency-of-Visits metric calculated for the crawlers and logs of our study. From this diagram, we observe that the Frequency of Visit of each crawler, as well as the relative ranking of crawlers based on their FoV values, vary from site to site. There is no evidence of any correlation between the FoV values observed and the popularity of a Web site or its geographic location. The four crawlers that belong to general search engines, however, visit all sites less than twice a month. CiteSeer visits the different sites at hand less frequently.

4.3. Coverage

The comparison of the number of resources discovered by a crawler on a particular Web site, with the resources sought by all the clients of this site, provides an estimate of the exhaustiveness of this crawler's visit on the particular site. Our conjecture is that requests issued by all clients represent a good approximation of discoverable resources available on a particular site.

We define accordingly the coverage of a Web site as the ratio of the distinct URL requests issued by a crawler over the distinct URL requests issued by all Web clients to the site at hand, during the same time-frame. The higher this ratio is, the more exhaustive is the crawler in its crawl of a site. The definition of the *coverage metric* m_{cov} follows:

Definition 3. (Resource Coverage metric) *The percentage of URL resources available at a Web site and retrieved by a Web crawler* $\mathcal{C}$*, within the time-frame captured by access log* L*, is defined as follows*

$$m_{\text{cov}}(L, \mathcal{C}) = \frac{N_{\mathcal{C}}^{d}(L)}{N^{d}(L)} \times 100 \tag{3}$$

where

- $N_{\mathcal{C}}^{d}(L)$ *is the number of requests issued by crawler* $\mathcal{C}$ *for distinct URL resources and recorded in access-log* L;

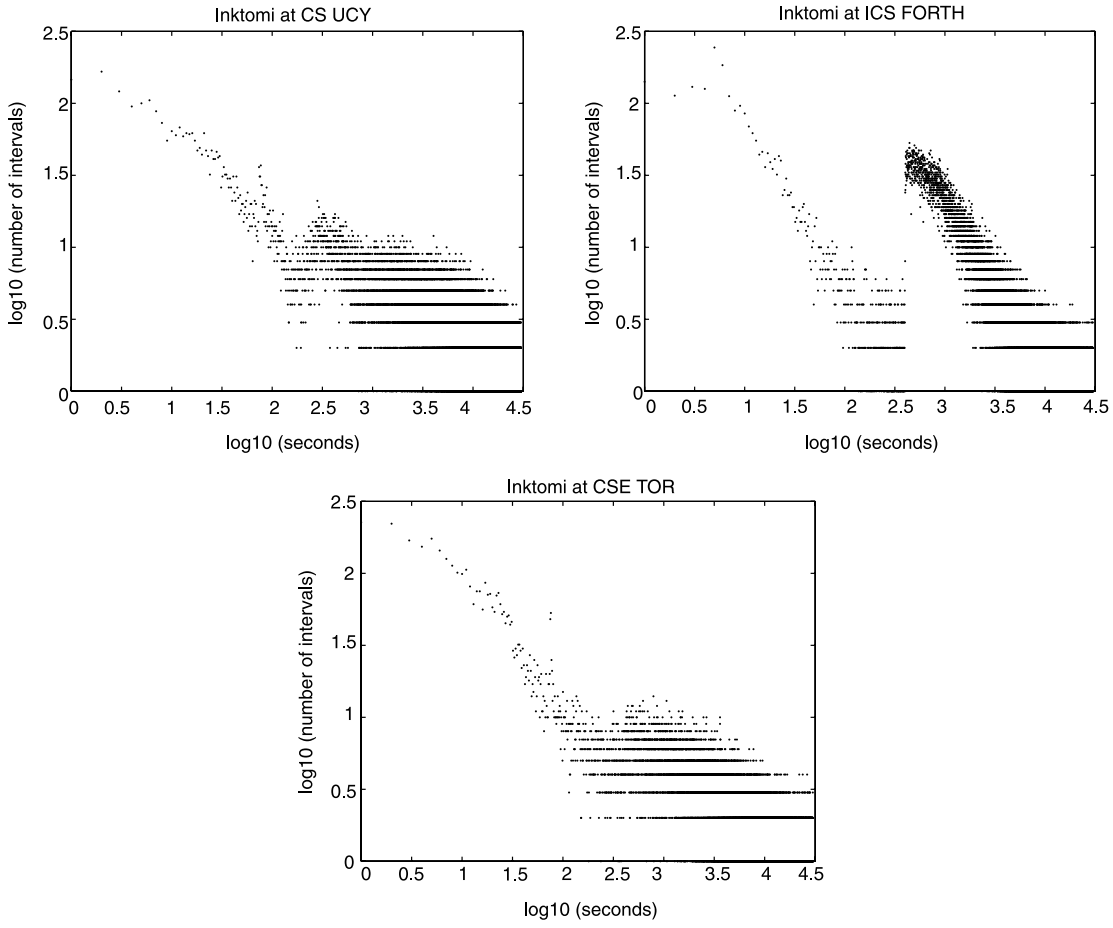

Fig. 12. Distribution of inter-arrival times for Inktomi crawlers.

- *$N^d(L)$ is the number of requests for distinct URL resources issued by all clients and recorded in access-log L.*

Fig. 16 (right) presents a diagram of our coverage measurements. From this plot, we can see that there is a wide variability in the coverage of different sites from most crawlers of our study, with the exception of Google and CiteSeer. Also, we can distinguish crawlers that are more exhaustive in their visits than others; for instance, Inktomi and AltaVista versus Google, FastSearch or CiteSeer.

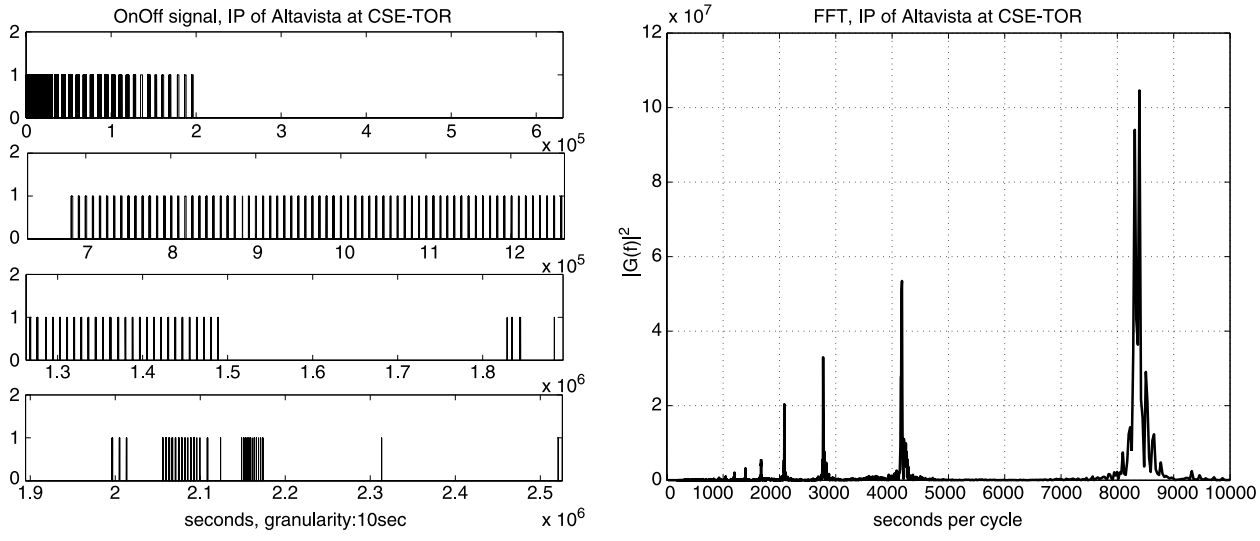

Fig. 13. ON-OFF Signal and its power spectral density of an AltaVista IP address hitting CSETOR.

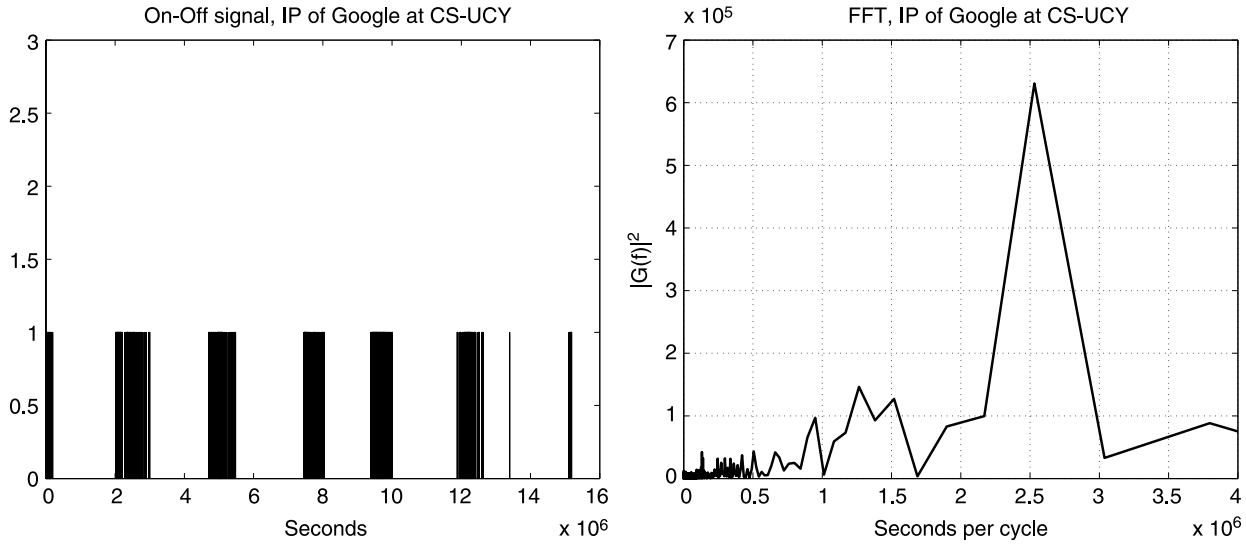

Fig. 14. ON-OFF Signal and its power spectral density of a Google IP address hitting CS-UCY.

5. Related work

Log analysis and Web characterization have been the target of intensive research in recent years, providing significant insights into Web usage and design [8,9,10,16,20–22]. Machine learning and data mining techniques have been applied to process logs in order to mine user profiles, communities of pages, patterns of use, and guide the improvement of Web design, the development of personalized sites, etc. [23–25]. Most characterization studies that we know of, however, focus on general Web traffic.

Very few studies have examined the behavior of Web crawlers as captured in Web-server access logs; these studies focus primarily on crawler detection rather than on crawler characterization. For instance, Almeida et al. proposed a set of heuristic criteria for identifying robots (crawlers and shopbots) in Web-server access logs [6]. They applied these heuristics for the identification of robots in a 15-day long access log of an online bookstore site. Subsequently, they assessed the impact of robot requests in Web caches and compared the behavior of crawlers and shopbots. In the context of their paper, the authors studied the distribution of interarrival times of crawler requests. They found evidence of periodic behavior and identified a log-normal distribution of interarrival intervals. Their remarks agree with our results on the periodic aspect of crawler visits. However, our study shows that interarrival times are heavy tailed and highly variable. The high variability observed in our work is due to the fact that we group together the requests of different crawling threads that belong to the same distributed crawler. Almeida et al. investigated also the popularity of resources referenced by crawlers; they found ‘quasi-horizontal regions’ indicating near-uniform referencing patterns for large groups of resources. This observation agrees with our remarks for some of the access-logs and crawlers of our study.

In Ref. [27], Tan and Kumar proposed a machine-learning technique to detect Web robots based on the navigational patterns in the click-stream data and applied their method on the University of Minnesota CS department server logs collected over a period of one month (January 1st to January 31st 2001). This paper, however, provides very limited information on the behavior of the crawlers studied and the characteristics of their traffic.

In Ref. [28], Ye, Lu and Li sought to minimize the impact of crawlers to the performance of busy Web-servers. The authors studied the hourly distribution of request arrivals of five crawlers (google, inktomi, baidu, webfountain

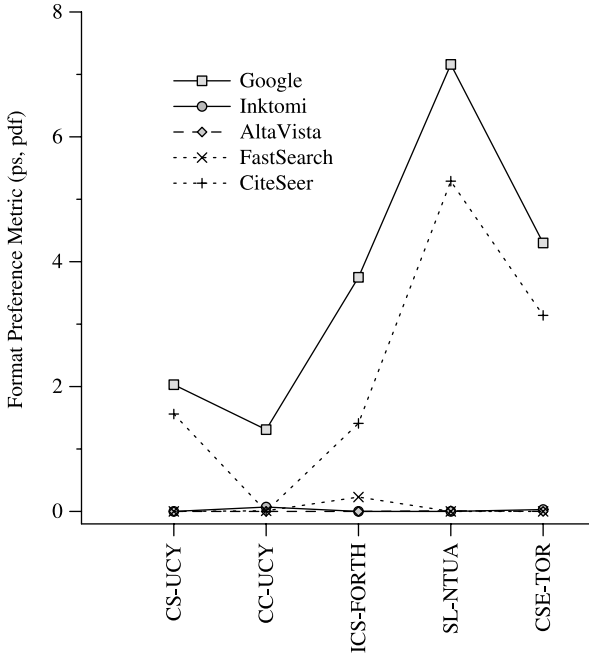

Fig. 15. Format preference metric for PS and PDF content.

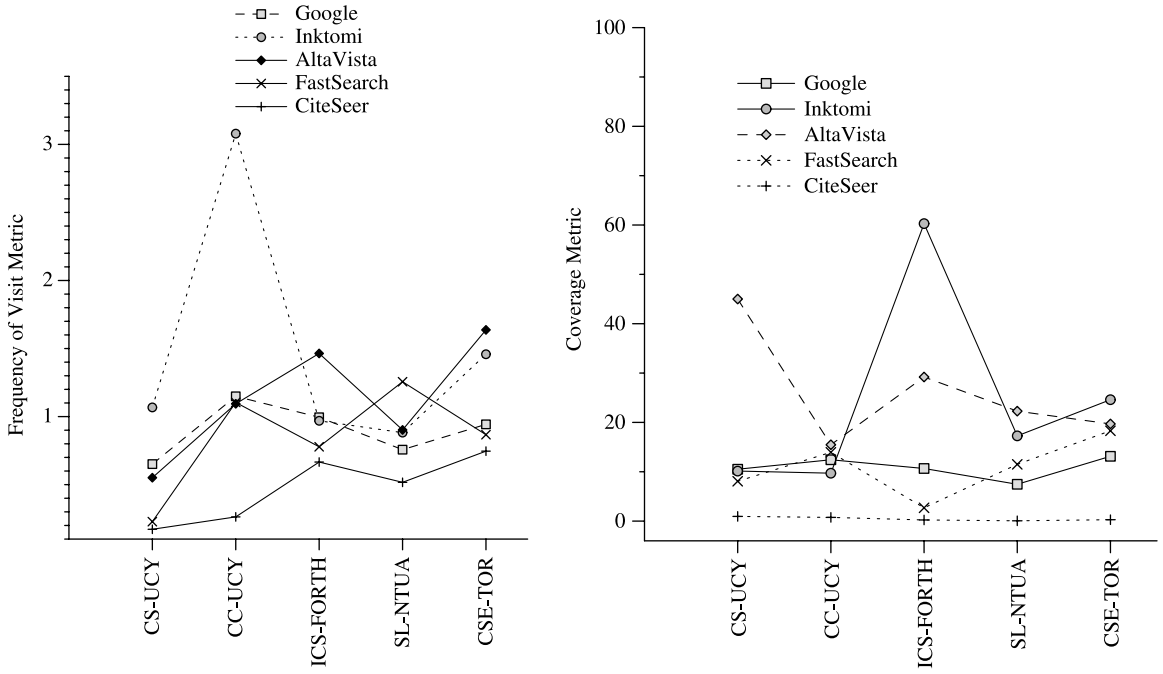

Fig. 16. Frequency of visit and coverage metrics.

and altavista), using a 6-month log from the Web servers of the China Education and Research Network (CERNET), which captures over 1 billion HTTP requests. They noticed that crawler visits often take place at peak hours of server activity. To address this problem, they proposed a crawling strategy that estimates the workload of a Web server just before starting a crawl of its pages, and schedules crawls during off-hours.

In contrast to the work presented in [6] and [27], the emphasis of our paper is on *crawler characterization* rather than on crawler detection. Therefore, we examine in more depth various aspects of Web-crawler behavior, focusing on crawlers that belong to five well-known search engines and using access-logs from five different sites in three countries, covering longer periods of time. The analysis presented in [28], on the other hand, focuses on crawler scheduling. The authors use logs from a single Web site, which is much busier than the ones used in our study. Nevertheless, the authors limit their investigation to the hourly spread of crawler requests at that site and do not provide any insights regarding the retrieved resources, inter-arrival times, the periodicity of crawls, and the HTTP traffic exchanged between the crawlers and the server. The motivation behind our analysis is to gain a significant insight into crawler behavior and to discover possible similarities with well-known statistical distribution functions. Apart from the findings being interesting in their own right, such an analysis is of great importance when one's ultimate goal is to separate robots from human users in logs.

6. Conclusions

In this paper, we presented a study of crawler behavior based on Web-server access logs from five different sites in three countries. Our logs capture the HTTP traffic of these sites for periods ranging from 42 days to 6 months at the beginning of year 2002. Based on these logs, we analyzed the activity of different crawlers that belong to four major, general-purpose search engines (Google, AltaVista, Inktomi, and FastSearch) and one major Digital Library and search engine for scientific literature (CiteSeer). Our analysis produced a number of insights regarding crawler traffic and crawling characteristics. In particular, we observe that:

1. Crawler activity has a noticeable impact on Web-server workload. It is important that this impact does not affect the Web-site performance, especially during periods of increased end-user activity. Other studies have shown that this is not the case, and proposed the detection of Web-server workloads by crawlers, as an approach for reducing crawling intrusiveness [28].
2. Crawler-induced HTTP messages carry *GET* requests at a percentage much higher than the general population of Web clients. Furthermore, crawlers that implement caching and employ conditional *GET*'s, receive *304* replies at a rate significantly higher than the general population of Web clients. Consequently, caching at the crawler-side can reduce significantly crawler-induced HTTP traffic.
3. Crawler requests result to a percentage of HTTP replies carrying error codes (with *4xx* numbers) at a rate nearly

double than what is observed for the general Web-client population. The application of intelligent techniques can help crawlers detect and avoid broken or erroneous links, resulting to an improved crawler efficiency.

4. As expected, crawlers seek text and HTML resources at a rate much higher than the general population of Web clients. Crawler interest for images is minimal. Crawlers that belong to search engines that index non-textual formats (postscript, PDF, etc.), however, fetch this type of resources more aggressively than the general Web-client population.
5. In contrast to observations for a Zipf-like concentration of HTTP requests to available Web resources, crawler-induced requests are not concentrated to a small subset of Web-site resources. Furthermore, crawlers distinguish Web-site resources into separate subsets, where the resources of different subsets are being visited with a different frequency. In other words, the distribution of the frequency of crawler requests across different resources (resource 'popu-popularity') is not Zipf.
6. HTTP replies to crawler requests exhibit a high variability in size; the same remark holds for successful responses to crawler requests that carry Web resources back to the crawler. The size of these messages can be modeled as a heavy-tailed hybrid Pareto and log-normal distribution. Average and median size of crawler-induced HTTP responses are smaller than those for the general Web-client population.
7. Inter-arrival times of crawler requests are highly variable and heavy-tailed. Periodicity properties can be investigated with the Fourier transform, which shows a periodic pattern in the timing of crawler visits upon a site.

Finally, we propose a set of very simple metrics that capture and describe qualitative characteristics of crawler behavior: the *Format Preference* metric, which describes the preference of a crawler on resources of a particular format; the *Frequency of Visit* metric, which represents the frequency of visits of a crawler on a Web site; and the *Resource Coverage* metric, which represents the exhaustiveness of a crawler's visit on a particular site. These metrics are used to derive conclusions on the strategies that different crawlers apply when crawling different sites. For instance, using the Format Preference metric we can easily see that AltaVista, Inktomi, and FastSearch do not retrieve postscript and PDF documents, in contrast to Google and CiteSeer. Also, that the frequency of visits of the five crawlers is not correlated either to the geographic location or the popularity of Web sites.

Our remarks provide a basis for developing techniques to detect Web crawlers automatically, to improve crawler design, and to reduce the impact of crawlers on Web-server performance.

Acknowledgements

This work was supported in part by the Planning Bureau of the Republic of Cyprus, through the WebC-MINE grant for Scientific Collaboration between Cyprus and Greece. The authors wish to thank professors Angelos Bilas of the University of Toronto, Vangelis Markatos of the University of Crete, Manolis Skordalakis of the National Technical University of Athens, and Mr Yanos Pitas of the University of Cyprus Computer Center for providing access to Web server access logs of their institutions.

References

[1] http://citeseer.nj.nec.com (last accessed Oct. 2003).
[2] http://www.altavista.com.
[3] http://www.fastsearch.com.
[4] http://www.google.com.
[5] http://www.inktomi.com.
[6] V. Almeida, D. Menasce, R. Riedi, F. Peligrinelli, R. Fonseca, W. Meira Jr, Analyzing Web Robots and Their Impact on Caching. In *Proceedings of the Sixth International Workshop on Web Caching and Content Distribution*, June 2001, pp. 299–310.
[7] A. Arasu, J. Cho, H. Garcia-Molina, A. Paepcke, S. Raghavan, Searching the Web, ACM Transactions on Internet Technology 1 (1) (2001) 2–43.
[8] M. Arlitt, T. Jin, Workload Characterization of the 1998 World Cup Web Site. Technical Report HPL-1999-35R1, Hewlett-Packard Laboratories, September 1999.
[9] M. Arlitt, C.L. Williamson, Web Server Workload Characterization: The Search for Invariants. In *Proceedings of the 1996 Sigmetrics Conference on Measurement and Modeling of Computer Systems*, ACM, May 1996, pp. 126–137.
[10] P. Barford, A. Bestavros, A. Bradley, M. Crovella, Changes in Web client access patterns: Characteristics and caching implications, World Wide Web (special issue on Characterization and Performance Evaluation) 2 (1999) 15–28.
[11] P. Barford, M. Crovella, Generating Representative Web Workloads for Network and Server Performance Evaluation. In *Proceedings of the 1998 Sigmetrics Conference on Measurement and Modeling of Computer Systems*, ACM, 1998, pp. 151–160.
[12] L. Breslau, P. Cao, L. Fan, G. Phillips, S. Shenker, Web Caching and Zipf-like Distributions: Evidence and Implications. In *Proceedings of the 1999 IEEE Infocom Conference*, IEEE, March 1999, pp. 126–137.
[13] S. Brin, L. Page, The anatomy of a large-scale hypertextual (web) search engine, Computer Networks and ISDN Systems 30 (1–7) (1998) 107–117.
[14] S. Chakrabarti, M. van den Berg, B. Dom, Focused Crawling: A New Approach to Topic-Specific Web Resource Discovery. In *Proceedings of the 8th World Wide Web Conference*, Elsevier, Toronto, 1999, pp. 545–562.
[15] M. Crovella, Performance characteristics of the world-wide web in: C. Lindemann, G. Haring, M. Reiser (Eds.), Performance Evaluation: Origins and Directions, Springer, 1999, pp. 219–233.
[16] M.E. Crovella, A. Bestavros, Self-Similarity in World Wide Web Traffic. Evidence and Possible Causes. In *Proceedings of the 1996 Sigmetrics Conference on Measurement and Modeling of Computer Systems*, ACM, 1996, pp. 160–169.
[17] B. Davison, D. Brian, Davison's Web Caching and Content Delivery Resources 2002 http://www.webcaching.com.
[18] M. Dikaiakos, A. Stassopoulou, L. Papageorgiou, Characterizing Crawler Behavior from Web Server Access Logs. Technical Report TR-2002-4, Department of Computer Science, University of Cyprus, November 2002.

[19] M.D. Dikaiakos, D. Zeinalipour-Yazti, A distributed middleware infrastructure for personalized services, Computer Communications 27 (15) (2004) 1464–1480.
[20] A. Feldmann, Characteristics of TCP connection arrivals in: K. Park, W. Willinger (Eds.), Self-Similar Network Traffic and Performance Evaluation, Wiley, 2000.
[21] S. Gribble, E. Brewer, System Design Issues for Internet Middleware Services: Deductions from a Large Client Trace. In *Proceedings of the 1997 Usenix Symposium on Internet Technologies and Systems. (USITS-97)*, 1997.
[22] B. Krishnamurthy, J. Rexford, Web protocols and practice, Addison-Wesley, 2001.
[23] M. Levene, A. Poulovassilis, Web dynamics, Software Focus 2 (2) (2001) 60–67.
[24] G. Paliouras, C. Papatheodorou, V. Karkaletsis, C. Spyropoulos, Clustering the Users of Large Web Sites into Communities. In *Proceedings of the International Conference on Maching Learning (ICML)*, 2000, pp. 719–726.
[25] M. Perkowitz, O. Etzioni, Towards adaptive Web sites: Conceptual framework and case study, Artificial Intelligence 118 (2000) 245–275.
[26] ACM SIGCOMM. The Internet Traffic Archive. http://ita.ee.lbl.gov.
[27] P.-N. Tan, V. Kumar, Discovery of web Robot sessions based on their navigational patterns, Data Mining and Knowledge Discovery 6 (1) (2002) 9–35.
[28] S. Ye, G. Lu, X. Li, Workload-aware Web Crawling and Server Workload Detection. In *Network Research Workshop, 18th Asian Pacific Advanced Network Meeting (APAN 2004)*, July 2004. www.cn.apan.net/cairns/NRW/ (last accessed, Dec. 2004).
[29] D. Zeinalipour-Yazti, M. Dikaiakos, Design and implementation of a distributed crawler and filtering processor in: A. Halevy, A. Gal (Eds.), Proceedings of the Fifth International Workshop on Next Generation Information Technologies and Systems (NGITS 2002), volume 2382 of Lecture Notes in Computer Science, Springer, 2002, pp. 58–74.