The Stretched Exponential Distribution of Internet Media Access Patterns

Lei Guo[1], Enhua Tan[2], Songqing Chen[3], Zhen Xiao[4], and Xiaodong Zhang[2]

[1]Yahoo! Inc.
Sunnyvale, California, USA
lguo@yahoo-inc.com

[2]The Ohio State University
Columbus, Ohio, USA
{etan, zhang}@cse.ohio-state.edu

[3]George Mason University
Fairfax, Virginia, USA
sqchen@cs.gmu.edu

[4]Peking University
Beijing, China
xiaozhen@net.pku.edu.cn

ABSTRACT

The commonly agreed Zipf-like access pattern of Web workloads is mainly based on Internet measurements when text-based content dominated the Web traffic. However, with dramatic increase of media traffic on the Internet, the inconsistency between the access patterns of media objects and the Zipf model has been observed in a number of studies. An insightful understanding of media access patterns is essential to guide Internet system design and management, including resource provisioning and performance optimizations.

In this paper, we have studied a large variety of media workloads collected from both client and server sides in different media systems with different delivery methods. Through extensive analysis and modeling, we find: (1) the object reference ranks of all these workloads follow the stretched exponential (SE) distribution despite their different media systems and delivery methods; (2) one parameter of this distribution well characterizes the media file sizes, the other well characterizes the aging of media accesses; (3) some biased measurements may lead to Zipf-like observations on media access patterns; and (4) the deviation of media access pattern from the Zipf model in these workloads increases along with the workload duration.

We have further analyzed the effectiveness of media caching with a mathematical model. Compared with Web caching under the Zipf model, media caching under the SE model is far less effective unless the cache size is enormously large. This indicates that many previous studies based on a Zipf-like assumption have potentially overestimated the media caching benefit, while an effective media caching system must be able to scale its storage size to accommodate the increase of media content over a long time. Our study provides an analytical basis for applying a P2P model rather than a client-server model to build large scale Internet media delivery systems.

Categories and Subject Descriptors

C.2.5 [**Computer Communication Networks**]: Local and Wide Area Networks—*Internet*

General Terms

Measurement, Performance

Keywords

Traffic analysis, Modeling, Multimedia

PODC'08, August 18–21, 2008, Toronto, Ontario, Canada.

1. INTRODUCTION

Internet media traffic has kept increasing. It is reported that in the recent past, video traffic on the Internet doubles every 3-4 months [1]. Different from traditional Web traffic, Internet media traffic can be delivered through a variety of approaches, such as streaming, pseudo streaming, overlay multicast, and P2P techniques, in addition to the commonly used Web downloading. Unlike the widely accepted Zipf-like access pattern of Web traffic [12], where text-based content was dominant, existing studies on media traffic are largely workload specific, and the observed access patterns are often different from or even conflict with each other, due to the variety of media delivery systems and the diversity of media content. For example, Chesire et al. [15] and Yu et al. [32] report that the access pattern of streaming media is Zipf-like in a university campus and in a VoD system, respectively, while Acharya et al. [9] and Cherkasova et al. [14] find that it is not Zipf-like in a multicast-based Media-on-Demand server and in an enterprise server, respectively. For live streaming, Veloso et al. [29] report user interests are object driven and follow a Zipf-like profile, while Sripanidkulchai et al. [27] report the popularity of live media programs hosted by Akamai CDN follows a 2-mode Zipf distribution. For P2P media systems, Gummadi et al. [19] report that the access pattern of media workload in KaZaa system collected in a campus network is not Zipf-like, while Iamnitchi et al. [24] report that it is Zipf-like in another campus network. For the access pattern of user generated video contents such as YouTube video [2], a study of requests in a campus network by Gill et al. [17] reports it is Zipf-like while a server log based study by Cha et al. [13] reports a significant deviation from the Zipf model. As a result, a number of models have been proposed to characterize Internet media access patterns, such as the *Zipf with exponential cutoff* model [13], *fetch-at-most-once* model [19], *generalized Zipf-like distribution* model [28], and others.

The object access pattern has a significant impact on the locality of references and the performance of caching in Internet systems. However, due to a limited number of workloads (typically one or two in each study) and a constrained scope of media traffic (e.g., enterprise server logs or requests from a campus network), the analyses in these studies may not be sufficient to identify a general media access pattern, which is important for traffic engineering on the Internet and is critical to design, benchmark, and evaluate Internet media distribution systems. In reality, although many algorithms and systems for media caching/proxying have been proposed, including commercial products such as Helix Universal Proxy [5] and Microsoft Windows Media Proxy [7], few of them are practically used. Instead, due to the high quality requirement and resource demand for media delivery, current media systems tend to over-supply or over-utilize system and bandwidth resources for user satisfaction [23]. Thus, it is highly desirable to have an insightful understanding of media access patterns for both media system designers and network administrators.

In this study, we have analyzed a wide variety of media workloads on the Internet. The workloads were collected from both the client side and the server side in Web, VoD, and P2P environments between 1998 and 2006, where the media content is delivered using Web/P2P downloading and unicast/multicast streaming, through P2P clients, enterprise servers, and CDNs. The duration of these workloads ranges from a few days to more than two years and the user population ranges from several thousands to more than one hundred thousand. The number of client requests ranges from tens of thousands to hundreds of millions, the number of objects in each workload ranges from hundreds to hundreds of thousands, and the median of file sizes in each workload ranges from a few megabytes to several hundred megabytes.

Through extensive analysis, we find that despite the different media systems and delivery methods used, the reference ranks of media objects in all these workloads can be well fitted with the stretched exponential (SE) distribution. This distribution has two parameters. We find that one parameter well characterizes the media file sizes, the other parameter well characterizes the aging of media accesses. We also analyze factors that may affect the observed media access patterns, such as redundant traffic filtered by a cache and extraneous traffic introduced through an ad server, and show how a biased measurement can lead to a Zipf-like observation on media access patterns. We further analyze the evolution of media access patterns in media systems, and find the deviation of media reference rank distribution from the Zipf model increases along with the duration of the workload.

We have further proposed a mathematical model to analyze the performance of media caching systems with the stretched exponential distribution. Compared with the caching of Web objects, whose access pattern is Zipf-like, caching of media objects is far less effective, unless the cache space is enormously large. Our study further shows that with more requests to media objects over time, there is a great potential to improve the performance of client-side caching. However, this improvement may take months to years and consumes huge amount of storage. This implies that a performance-effective and cost-efficient media caching system should be capable of scaling its storage size with the increase of its workload size over a long time. Our study provides an analytical basis for applying a P2P model instead of client-server model to build large scale Internet media delivery systems.

The remainder of the paper is organized as follows. Section 2 summarizes the workloads we use. We present the stretched exponential model of media reference rank distributions in Section 3, and study the physical meanings of this access pattern model in Section 4. We analyze the impact of the stretched exponential distribution on media caching performance in Section 5, and make concluding remarks in Section 6.

2. WORKLOAD DESCRIPTION

In this study, we analyze a total of sixteen media workloads collected from both client and server sides in different media systems with different delivery methods, as outlined in Table 1. By "client side", we mean the client network where user requests are sent out. Table 1 also summarizes the workload duration, number of requests, number of clients, and number of objects for each workload. Among these workloads, six of them were collected by ourselves. We also have the access to two workloads available in public sources or provided by our research collaborators. The remaining eight, with an asterisk before the workload name in Table 1, are extracted from the figures in the published papers due to the unavailability of original data. These sixteen workloads are classified into four categories based on different media delivery environments, summarized as follows.

The first category is "Web media". These workloads contain media requests in the Web environment, where media objects are embedded or linked in Web pages. On these Web pages, the majority are still common Web objects such as text, graphics, and html files, etc. In this category, HPC-98 (extracted from Figure 7(a) of paper [14]) contains server logs collected from the HP Corporate media server, which hosts the streaming media objects accessed through the Web pages of the HP corporation. Similarly, HPLabs-99 (also extracted from Figure 7(a) of paper [14]) contains the logs of streaming servers hosting HP Labs media content, accessed through the HPLabs Web site. Similar to HPC-98, ST-SVR-01 is the log of a large enterprise streaming server, accessed through the company's Web site. We collected PS-CLT-04 and ST-CLT-04 from a large cable network hosted by a major ISP in the United States. Workload ST-CLT-05 was collected in a subdomain of the same network. PS-CLT-04 is a media workload of Web downloading and pseudo streaming, which includes the first IP packets of HTTP downloading for Windows, RealNetworks, and QuickTime media files. ST-CLT-04 and ST-CLT-05 are RTSP (RFC 2326) and MMS (Microsoft's proprietary streaming protocol) streaming media traces collected with a similar method to that by Chesire et al. [15].

The second category is "VoD media". Different from a Web site with a few media objects, a VoD system provides an integrated environment for dedicated media services, though it is often Web-based too. mMoD-98 contains logs of a multicast-based Media-on-Demand video server supporting VCR functions, where the major contents are video lectures and movies (extracted from Figure 6 of paper [9]). CTVoD-04 contains logs collected from streaming servers of a large VoD system deployed by China Telecom, where the major contents are TV shows and movies (extracted from Figure 13 of paper [32]). These two VoD systems host media content themselves. We collected IFILM-06 from the IFILM Web site [6], which provides the weekly click numbers of Web pages for IFILM video clips, most of which are short movie trailers. These video objects are served by a CDN via streaming. We have also collected YouTube-06 from its Web site [2] by crawling summary pages of YouTube video, where the total number of requests of each clip (for the entire up time of YouTube site) was published (the methodology is similar to that in [13]).

The third one is "P2P media", collected from two kinds of P2P systems. In KaZaa networks, users exchange files with each other, while in a BitTorrent swarm, users exchange chunks of the same file. KaZaa-02 is a large file transferring (larger than 100 MB, typically video files) workload over KaZaa networks, collected in a university campus (extracted from Figure 5 of paper [19]). KaZaa-03 is extracted from Figure 5 of paper [24], which includes music files, video clips, and movie files of different sizes. BT-03 contains data collected from two BitTorrent tracker sites, where most of the files are large videos and DVD movies [11].

The above three categories are all on-demand media workloads. We have also analyzed media workloads of live streaming and theater environments, categorized as "other" in our workload set. Workload Akamai-03 consists of the references to live streaming media programs hosted by Akamai CDN, extracted from Figure 3 of paper [27]. Workload Movie-02 contains the 2002 U.S. movie box office ticket sales, extracted from Figure 7(d) of paper [19]. Workload IMDB-02 is the cumulative number of votes for top 250 movies in Internet Movie Database (IMDB), which was downloaded from the IMDB Web site [3].

Client side streaming media workloads often contain *extra-*

Table 1: Workload Summary

System Type	Workload Name	Delivery Method	Workload Duration	Collection Time	Num. of Requests	Num. of Objects	Num. of Clients	Median File Size	c
Web Media	*HPC-98	streaming	29 months	11/98-04/01	666,074	2,999	131,161	14 MB	0.22
	*HPLabs-99	streaming	21 months	07/99-04/01	14,489	412	2,482	120 MB	0.3
	ST-SVR-01	streaming	122 days	04/01-07/01	169,414	2,260	41,709	15 MB	0.2
	PS-CLT-04	downloading	9 days	08/04	196,621	53,383	6,276	1.5 MB	0.2
	ST-CLT-04	streaming	9 days	09/04	61,889	18,511	4,751	2 MB	0.2
	ST-CLT-05	streaming	11 days	06/05	54,984	18,634	6,238	4.5 MB	0.2
VoD Media	*mMoD-98	multicast	194 days	08/97-03/98	–	139	–	125 MB	0.55
	*CTVoD-04	streaming	219 days	05/04-12/04	21 million	6,700	150,000	300 MB	0.4
	IFILM-06	streaming	16 weeks	03/06-07/06	62,228,780	11,872	–	2.25 MB	0.15
	YouTube-06	pseudo stream	all-time	10/06	692,343,054	3,981,654	–	3.4 MB	0.17
P2P	*KaZaa-02	exchange	203 days	05/02-12/02	98,997,622	633,106	24,578	300 MB	0.45
	*KaZaa-03	exchange	5 days	01/03	976,184	116,509	14,404	5 MB	0.14
	BT-03	swarming	48 days	10/03-12/03	256,802	2,453	45,058	636 MB	0.52
Other	*Akamai-03	live stream	3 months	10/03-01/04	70 million	5,000	–	–	0.2
	*Movie-02	–	1 year	year 2002	–	250	–	–	0.65
	IMDB-06	–	all-time	06/06/2006	–	250	–	–	1.15

neous media objects that are pushed to the user mandatorily no matter the user wants them or not. In commercial streaming media systems, for advertisement purposes, normally when a user clicks the meta file link of a media object on a Web page, the media server requests a link of *ad clip* from an ad server, inserts it to a dynamically generated meta file, and then sends the meta file to the client player. Furthermore, some media servers may insert the link of *flag clip*—a small video or audio object (usually less than 5 seconds) that plays a silent audio or blank video, a static logo image, or a quick animation—before the URL of a requested media object and/or between two subsequent requested media files, in the dynamically generated meta file. Ad clips are usually served by dedicated ad servers outside the content system, while flag clips are usually served by the same server serving program content. However, both ad and flag media traffic is *extraneous* to users of the media system, and do not reflect the real user access pattern. Our study shows that although ad and flag clips only account for a small percentage of media traffic, they usually have significantly higher access rates than normal content objects. In our collected client side streaming workloads ST-CLT-04 and ST-CLT-05, we identified and removed ad and flag clip requests by matching the URLs in RTSP commands with keywords such as "ads" and "logo", "getnext", "next", which indicate the purpose of these objects, as well as by viewing/listening to the video/audio content of these objects. For server side media workloads, the requests of ad media are not recorded in server logs, since media servers and ad servers are usually separate. Meanwhile, not every media system uses flag clips, and the number of flag clips in a media system is very small (usually 1 or 2).

3. THE REFERENCE RANK MODEL OF INTERNET MEDIA OBJECTS

3.1 The stretched exponential of Internet media traffic

We use the rank-ordering technique to analyze the Internet media access pattern. Figures 1, 2, and 3 show the reference rank distributions of media objects in Web, VoD, and P2P media systems, respectively[1]. In each figure, the x coordinate represents the reference rank of each object, plotted in log scale, while the y coordinate represents the number of references to this object, plotted in both log scale (marked on the right of y-axis) and a powered scale (by a constant c, as marked on the left of y-axis). We call the combination of log scale in x and powered scale in y as the stretched exponential (SE) scale.

These figures show that in log-log scale, the reference rank distributions of all these workloads have a *fat* head and a *thin* tail, which cannot be fitted with a straight line, indicating they are not Zipf-like. In particular, many of them deviate from a straight line significantly, such as Figures 2(a), 2(b), and 3(c). However, by selecting a proper constant c, all these workloads can be well fitted with a straight line in SE scale. Such a distribution is called a *stretched exponential distribution.*

To evaluate the stretched exponential fit, we compute the coefficient of determination of the fitting result of each workload, R^2. As marked in the figures, R^2 is very close to 1 for all workloads. For workloads with raw data accesses, χ^2 tests are conducted to check the goodness of fits. The stretched exponential fits are accepted while Zipf-like fits are rejected (see Appendix B of [20]). For long term workloads with timestamps of requests, including ST-SVR-01, BT-03, and IFILM-06, stretched exponential fits and χ^2 tests are further conducted on the reference rank distributions of objects requested in different durations (see Section 4.2).

The probability distribution of the stretched exponential distribution can be expressed as

$$P(X < x) = 1 - e^{-(\frac{x}{x_0})^c}, \tag{1}$$

where c and x_0 are constants. If we rank the N objects in the workload in descending order of their reference numbers y_i $(1 \leq i \leq N)$, we have $P(y \leq y_i) = i/N$. So the rank distribution can be expressed as follows

$$y_i^c = -a \log i + b \ \ (1 \leq i \leq N), \tag{2}$$

where $a = x_0^c$ and $b = y_1^c$. Since the minimum number of references to an object is 1, we can assume $y_N = 1$ when the number of objects in the workload, N, is large enough[2]. Thus

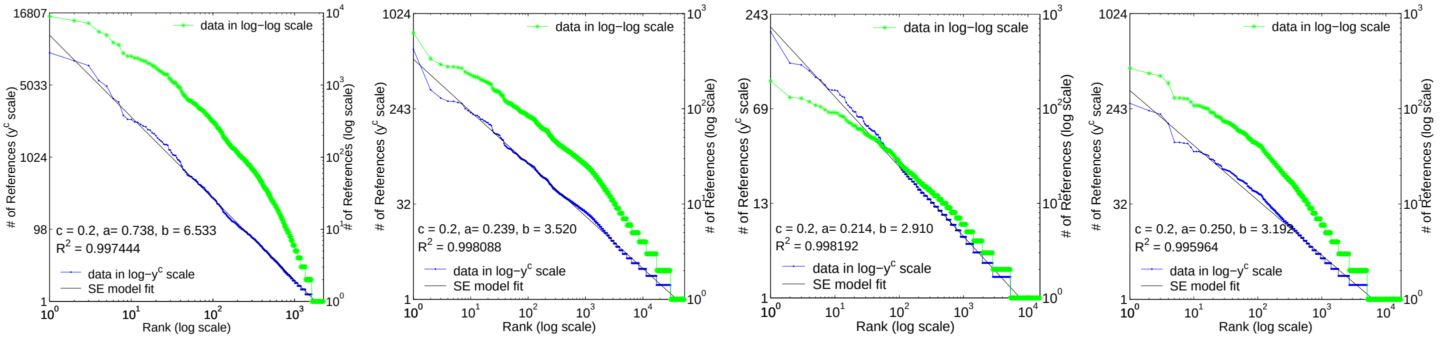
$$b = 1 + a \log N. \tag{3}$$

An SE distribution curve is a straight line in SE scale. Since b is a normalization parameter, the shape of an SE distribution is determined by c, the *stretch factor* of y coordinate, and a, the minus of the slope of the straight line in SE scale. An SE distribution has a finite mean value (denoted as $\langle x \rangle$)

$$\langle x \rangle = \int_0^\infty x p(x) dx = x_0 \Gamma(1 + \frac{1}{c}), \tag{4}$$

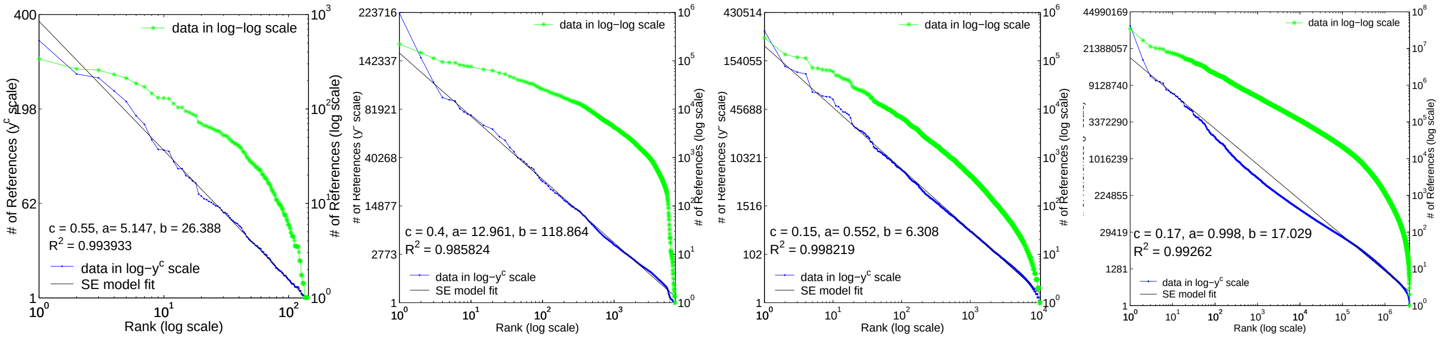
[1] The extraneous traffic in ST-CLT-04 and ST-CLT-05 has been removed. The figures of workloads HPC-98, HPLabs-99, longer duration of workload IFILM-06, Akamai-03, Movie-02, and IMDB-06 are presented in Figure 16 of Appendix A.

[2] As shown in Appendix A, for server side workloads, it is possible that $y_N > 1$ due to the small number of objects.



(a) ST-SVR-01 (b) PS-CLT-04 (c) ST-CLT-04 (d) ST-CLT-05

Figure 1: Reference rank distributions of media objects in Web systems



(a) mMoD-98 (b) CTVoD-04 (c) IFILM-06 (first 7 days) (d) YouTube-06 (all time)

Figure 2: Reference rank distributions of media objects in VoD systems

where $p(x) = c\frac{x^{c-1}}{x_0^c}e^{-(\frac{x}{x_0})^c}$ is its probability density function and $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1}e^{-t}dt$ is the gamma function. However, for workloads with a limited number of objects, the deviation of the average number of references to objects (denoted as $\langle y\rangle$) from Equation 4 can be non-trivial, especially when $\langle y\rangle$ is small (since the minimum number of references is 1, not zero). A better estimation is

$$\begin{aligned}\langle y_{se}\rangle &= \lim_{N\to\infty}\frac{1}{N}\sum_{i=1}^{N}(1-a\log\frac{i}{N})^{\frac{1}{c}}\\ &= \lim_{N\to\infty}\int_{\frac{1}{N}}^{1}(1-a\log x)^{\frac{1}{c}}dx\\ &= e^{\frac{1}{a}}x_0(\Gamma(1+\frac{1}{c})-\gamma(1+\frac{1}{c},\frac{1}{a})),\end{aligned}\tag{5}$$

where $\gamma(\alpha, x) = \int_0^x t^{\alpha-1}e^{-t}dt$ is the lower incomplete gamma function. Throughout this paper, the modeling analysis is mainly based on Equation 5, while Equation 4 is only used as an approximation for simplicity.

The stretched exponential distribution has been used to describe many phenomena in nature and economy that do not follow power law [25]. Although it is still empirical, in the subsequent parts of this paper, we will show that the stretched exponential model of Internet media reference rank distributions has clear physical meanings. Different from the systems studied in [25], we analyze a number of media systems with different sizes of media files and in different lengths of durations, and find the relations between these factors and the parameters of the stretched exponential model, which further supports the validity of this model.

3.2 Factors that may affect media access pattern observations

In addition to user activities, the computing and networking systems that an entire media delivery procedure involves may affect the measurement results of media object reference rank distributions by filtering redundant traffic (e.g., through a proxy) or introducing extraneous traffic (e.g., through an ad server). In this section, we study how these factors affect the *observed* media access patterns, and show how a biased measurement can lead to a Zipf-like observation as reported in some studies before.

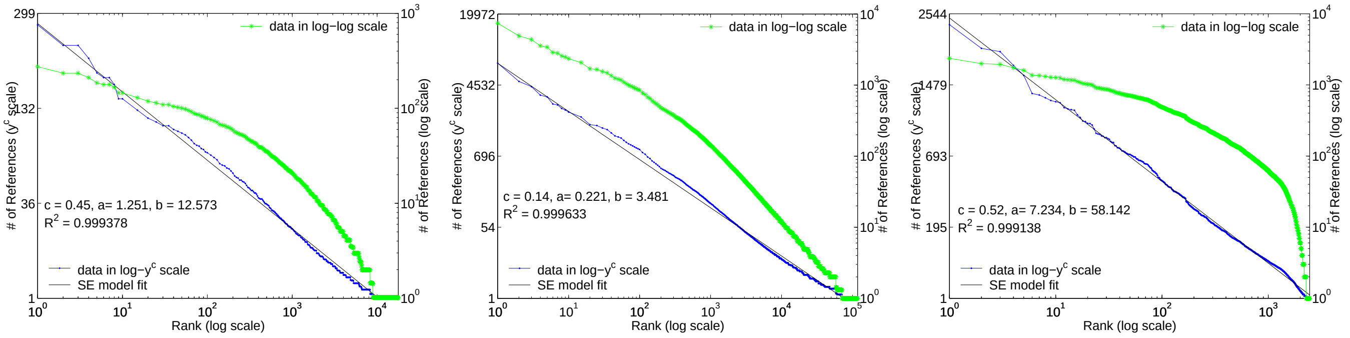
3.2.1 Effect of extraneous traffic

As we have shown in Section 2, the client side streaming media workloads often contain extraneous traffic such as ad and flag media clips, which do not reflect the real user access pattern and may significantly affect the object reference rank distribution. Without removing extraneous traffic, the left of Figure 4 shows that the log-log plot of the reference rank distribution of media objects in workload ST-CLT-04 can be well fitted with a straight line, indicating a Zipf-like distribution (see Equation 14 of Section 5.1.1) with a skewness factor $\alpha \approx 0.71$. Similarly, Chesire et al. [15] also find that the reference rank distribution of a streaming media workload collected in University of Washington follows a Zipf-like distribution [3].

However, as shown in Figure 1(c), after removing the extraneous traffic (31% requests in the workload), the distribution can be well fitted with a stretched exponential model, while the log-log plot has a clear curvature with a fat head and a thin tail. Since we use a similar method to collect streaming media traffic as that used in study [15], it is possible that the workload used in [15] also contained some extraneous media traffic, which causes the observed Zipf-like distribution.

Even with extraneous media traffic, it does not seem to be true that a Zipf-like distribution can always be observed on the access pattern of media objects: we find the extraneous traffic in different workloads varies. As shown in the right

[3] See Figure 8 of paper [15]. The skewness factor (the minus of the slope of log-log plot) estimated from the figure is about 0.67 (instead of 0.47 as reported), quite close to that of workload ST-CLT-04 (0.71).



(a) KaZaa-02 (b) KaZaa-03 (c) BT-03

Figure 3: Reference rank distributions of media objects in P2P systems

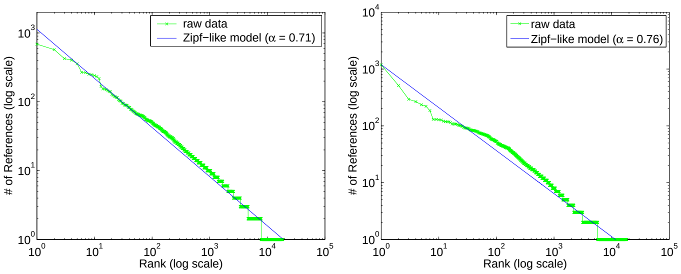


Figure 4: Zipf-like fit for workload with extraneous traffic (Left: ST-CLT-04, Right: ST-CLT-05)

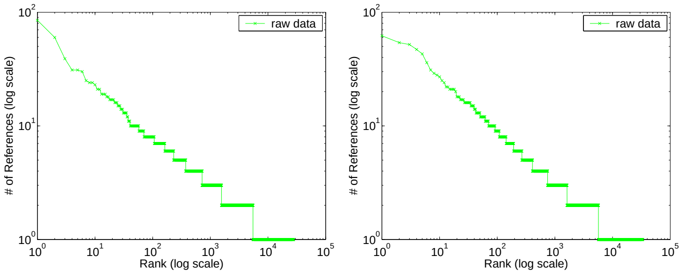


Figure 5: Number of requests to each object by the same user (Left: ST-CLT-04, Right: ST-CLT-05)

of Figure 4, the raw data in the ST-CLT-05 workload (with ad and other extraneous media traffic) does not fit the Zipf-like distribution well. Analyzing the traces we collected, we find the reason is due to the changes of ad and flag media clips in the workloads. In workload ST-CLT-05, we find the percentage of extraneous media traffic from one of the major advertisement media providers, yahoo.com, was significantly reduced compared with ST-CLT-04, and the total number of ads requests only accounts for 13% (still non-trivial) of all media requests in workload ST-CLT-05. Furthermore, in workload ST-CLT-05, we find several large media providers, such as MSNBC, merged their flag and logo clips into media content objects when the objects were authored, thus the number of requests of flag clips was also significantly reduced.

3.2.2 *Caching effect*

Previous studies on Web workloads have also shown that the initial part of the reference rank distribution of objects can be lower than what the Zipf model predicts, which looks similar to our SE observation on media workloads. Williamson et al. [31] point out this is due to the caching effect: popular Web objects are likely to be cached by Web browsers or proxies, hence, subsequent requests may not reach the server. For Web traffic, usually caching only affects the initial part of the reference rank distribution (depending on the cache size), and the main body of the distribution curve is still a straight line in log-log scale. Similarly, as reported by Cherkasova et al. [14], for short term media workloads, the object reference rank distribution in log-log scale does not significantly deviate from a straight line except the initial part. Thus, it is important to figure out whether the stretched exponential distribution of media access patterns we observed is due to the caching effect of media content.

However, for streaming workloads in this study, the caching effect has been considered. Although a media player can cache media files that have been delivered and played, it still sends a report to the server when the object is re-played in the local cache, in order to let the server generate a log entry [8]. Similarly, both Windows and RealNetworks media proxies also send a report to the server when the cached content is requested, so that the server can collect its object access information and generate a log entry for the request [7, 5]. Thus, server log based workloads include all requests to media objects, even for cached objects. For client side workloads we collected, we have carefully extracted the RTSP/MMS commands for cache-validation and log statistics to include local replay events. Thus, from the perspective of measurement methodology, server side workloads HPC-98, HPLabs-99, ST-SVR-01, CTVoD-04 and client side workloads ST-CLT-04 and ST-CLT-05 reflect the real access patterns of media users. Therefore, caching is not likely to be the reason that causes the observed stretched exponential access patterns in these workloads.

Meanwhile, caching effect is also observed among requests during multiple weeks for workload IFILM-06. Workload IFILM-06 records the number of clicks on the page of each media object published on the IFILM Web site. Since the Web page of a media object can be cached by a Web browser or a proxy, a reloaded page may not be counted as a page click. Thus, as shown in Figure 16(c) (see Appendix A), due to the accumulation of caching effect over time, the initial part (for the first 100 objects at most) of the reference rank distribution deviates from the SE model gradually, but the main body of distribution still follows the SE model. However, for short duration media workloads, the caching effect is trivial. For example, both the 7-day workload of IFILM-06 (Figure 2(c)) and the 9-day workload PS-CLT-04 (Figure 1(b), pseudo streaming via HTTP) are barely affected by client side caching.

3.2.3 *"Fetch-at-most-once" effect*

Similar to the caching effect, Gummadi et al. [19] find that P2P traffic is not Zipf-like and attribute this to the "fetch-at-most-once" effect: since typically media objects do not change over time, P2P users download them at most once (this effect is equivalent to have an unlimited cache on each client). Assuming the playback activities of P2P users follow a Zipf-like distribution, the authors show that the object reference rank distribution of corresponding downloading activities (workload KaZaa-02) is very close to the real reference rank distribution of P2P workloads by simulations.

However, this model can only explain the access pattern of P2P workloads and needs a free parameter to specify the number of users. For streaming media workloads, as we have presented in Section 3.2.2, the local replaying events due to caching are recorded in server logs and have been carefully measured in our client side network measurements. (1) For streaming media workloads of small objects, such as ST-SVR-01, ST-CLT-04, and ST-CLT-05, there is no "fetch-at-most-once" effect, because caching has been considered. These workloads do not follow the Zipf-like distribution, but stretched exponential. Furthermore, as shown in Figure 5, for the MMS traces in workloads ST-CLT-04 and ST-CLT-05 (extraneous traffic excluded), it is not rare that an object is requested by the same user multiple times (multiple computers shielded by NAT are identified with the technique in [23]). (2) Similarly, for server log based streaming media workloads of large video files, such as CTVoD-04, the "fetch-at-most-once" effect does not exist either. However, the access pattern of workload CTVoD-04 deviates from the Zipf model even more significantly than those small file workloads. Although "fetch-at-most-once" might be the case in P2P workloads, for large video objects, such as files in KaZaa-02, it is also reasonable to assume users will not repeatedly watch the same video, whether it is rented from a store or downloaded through P2P networks. Thus, it is very likely that user access patterns in P2P workloads of large video files reflect the real user playing activities quite well. This is also evidenced by the similar access patterns found in P2P downloading workload KaZaa-02 and VoD streaming workload CTVoD-04: both the median file sizes and stretch factors of these two workloads are very close. We will further analyze these two workloads and this similarity in Section 4.1.

3.3 Other media access pattern models

To our best knowledge, for most reported Zipf-like media access patterns in existing literature, the model fitting is quite rough. For example, in studies [18], [22], and [24], the distribution curve does not strictly follow a straight line in log-log scale. In study [32], although the head and waist of the distribution curve in log-log scale are roughly in a straight line, the tail of the distribution curve, which accounts for the majority of media objects, is far from the straight line. Our stretched exponential model can well fit or explain these workloads.

Besides these studies, a number of models have been proposed to describe or explain the deviation of media access patterns from the Zipf-like distribution. Most of them are still based on the Zipf model. For example, Cha et al. [13] use the Zipf with an exponential cutoff effect to describe the thin tail of media popularity distributions. In the paper, this effect is explained as preferential attachment with information filtering. The basic idea of preferential attachment mechanism is the "rich-get-richer" effect: If k users have fetched an object, the rate of other users fetching it is proportional to k. This model implies that a popular object can keep popular continuously, rather than become unpopular over time. For Web workloads, this argument is true: some pages, especially the first pages of Web portals and search engines, such as Yahoo!, Google, and MSN, can have a high popularity rank for a very long time. For example, according to Alexa Internet (`http://www.alexa.com`), the daily traffic rank of Yahoo! keeps number one from 2001 (earlier data are unavailable) until April, 2008.

Nevertheless, this argument is not valid for media objects. Web pages can be updated frequently to attract users and keep their popularity. In contrast, media objects are usually immutable. In a long term, a video object, no matter how popular it is, is unlikely to keep popular or become more popular with the passage of time and with the creation of new objects. On the contrary, many measurement studies have observed that the popularity of a media object becomes unpopular quickly [13, 19, 21]. For example, study [21] reports media popularity decreases with time exponentially. Thus, the "rich-get-richer" phenomenon reflected in Zipf and power law based models is not present in media objects.

We have compared the top 1, top 2, ..., top 100 popular objects of every week in a Web workload (collected by University of Calgary [10]) and a video workload (IFILM-06) for sixteen weeks. Figure 6 shows the total number of distinct objects in the union set of these sixteen weekly top N objects ($1 \leq N \leq 100$), for the Web and video workloads, respectively. We can see that the set of 16 weekly top N popular Web objects is much smaller than that of 16 weekly top N popular video objects. Particularly, the top one popular Web object never changes, but the top one popular video object changes every week. This indicates the number of references to a popular media object cannot be linearly accumulated with time. The absence of "rich-get-richer" effect can also be reflected by the evolution of media object popularity distributions in a long duration, which has been reported by study [14]. In Section 4, we will show that although in a short duration, the popularity distribution of media objects in a system might be skewed and look Zipf-like in log-log scale, the skewness tends to weaken rather than strengthen over time, so that finally the distribution looks unlike Zipf at all.

We have also compared the stretched exponential model with other two-parameter media access pattern models, such as Zipf-Mandelbrot model [26] and parabolic fractal model [16]. We find these models can only fit a small number of workloads, while SE fits all (see Appendix C of [20]).

4. THE DYNAMICS OF MEDIA REFERENCE RANK DISTRIBUTIONS

In this section, we analyze the reference rank distributions of media objects of different sizes in different durations, in order to further understand the stretched exponential access patterns in different media systems. Our analysis shows: (1) parameter c well characterizes the effect of media file sizes; (2) parameter a well characterizes the non-stationarity effect of media access aging; (3) the deviation of media reference rank distribution from the Zipf model increases with the length of workload duration.

4.1 Access patterns of different sized media files

Figure 7 shows the stretch factor c of each on-demand media workload (Web, VoD, or P2P) with its median file size (also see Table 1). In this figure, each point represents a single workload. Roughly distributed along a straight line, these points can be classified into three groups: for workloads with a median file size < 5 MB, the stretch factor is ≤ 0.2; for workloads with a median file size > 100 MB, the stretch factor is ≥ 0.3; for other workloads, the stretch factor is between 0.2 and 0.3. As shown in Table 1 (column 3, 9, and 10), in general, for media workloads delivered by similar systems and techniques, their stretch factors increase with their median file sizes. For example, the median file size of workload KaZaa-03 is 5 MB, and the median file size of workload KaZaa-02 is 300 MB. Their stretch factors are 0.14 and 0.45, respectively. For media workloads delivered by streaming techniques, the median file sizes of workloads IFILM-06, ST-CLT-05, HPLabs-99, and CTVoD-04 are 2.25 MB, 4.5 MB, 120 MB, and 300 MB, respectively, while their stretch factors in the SE model are 0.15, 0.2, 0.3, and 0.4, respectively.

Meanwhile, for workloads with similar median file sizes, in

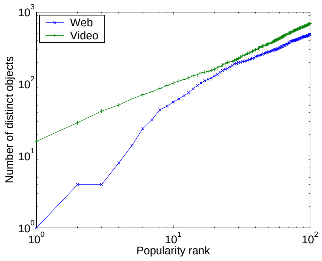


Figure 6: Number of distinct weekly top-N popular objects

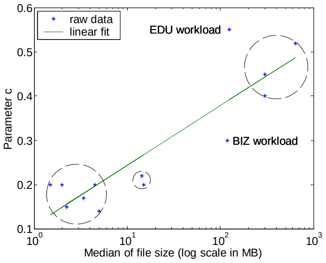


Figure 7: Stretch factor vs. median file size

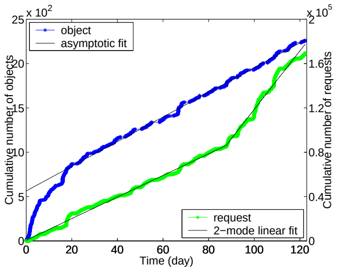


Figure 8: Media birth and access over time (ST-SVR-01)

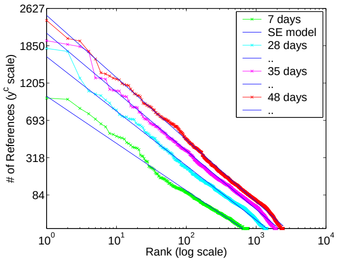


Figure 9: Evolution of reference rank distribution (BT-03)

most cases, the corresponding stretch factors are also similar regardless of the underlying media systems and delivery techniques. For example, workload KaZaa-02 is delivered by P2P networks while CTVoD-04 is delivered by VoD servers. However, they have similar median file sizes (both about 300 MB) and similar stretch factors (0.45 and 0.4, respectively). Workload KaZaa-03 is delivered by P2P networks while IFILM-06 is delivered by a CDN. Their median file sizes are both less than 5 MB, and their stretch factors are 0.14 and 0.15, respectively. Workload PS-CLT-04 is delivered by downloading while ST-CLT-04 is delivered by streaming. Both of their median file sizes are about 2 MB, and both of their stretch factors are 0.2. These cases indicate that in general, despite the different techniques and systems used for media delivery, the larger the median file size of a workload, the greater the stretch factor of its SE reference rank distribution.

Our analysis indicates that the reference rank distribution of a media workload highly depends on the median file size of the workload rather than on the delivery method and the underlying media system. Although the relationship between the median file size and the stretch factor of a media workload shown in Figure 7 has not been strictly formalized, the following factors may help us further understand the trend reflected in this figure. (1) It is unfair to compare files with similar sizes (in bytes) but with different encoding rates and compression ratios. The file length (in seconds) or user playback duration might be a more objective metric, but unfortunately we cannot use this metric because the relevant data are unavailable in some workloads. (2) Video and audio may have different access patterns and thus have different stretch factors. In our workloads, KaZaa-03 is MP3 music dominant, all Web media workloads have a combination of video and audio, and other workloads are video dominant. Due to the lack of related information in the workloads, we cannot separate video and audio in the analysis. (3) Different types of video content (e.g., entertainment, business, and educational content) may have different access patterns. In our study, HPLabs-99 is a business workload, where the content is hosted by an internal media server open for research employees in HP Labs only. Workload mMoD-98 is collected in an educational environment, where the content is a combination of video classes and movies. Other workloads are all entertainment content dominant. HPLabs-99 and mMoD-98 have similar median file sizes but the values of parameter c are 0.3 and 0.55, respectively. As shown in Figure 7 ("EDU" for mMoD-98 and "BIZ" for HPLabs-99), both workloads deviate from the straight line greater than other workloads.

4.2 Non-stationarity of media reference rank distributions

Studies have found that the access patterns in long term media workloads are not Zipf-like though they look Zipf-like in short term media workloads [14, 28]. In contrast, the access patterns in long term Web workloads are still Zipf-like even for workloads of several months to one year [30]. In this section, we study the stationarity of media reference rank distributions, and show how media access aging affects parameter a, the minus of the slope of SE plot.

To characterize the dynamics of a media system, we consider the birth rate of new objects (denoted as λ_{obj}) and the request rate of all objects (denoted as λ_{req}) first. We have analyzed media accesses over time for long duration workloads ST-SVR-01, BT-03, and IFILM-06 as well as short duration workloads PS-CLT-04, ST-CLT-04, and ST-CLT-05 (other workloads have no timestamps or temporal information of requests). In this section we focus on long duration workloads in order to study the trend of media access evolution. For workloads BT-03 and IFILM-06, we find that the cumulative number of requests increases with time linearly in a coarse time granularity, indicating that the media request rates in corresponding systems are roughly constants. For workload ST-SVR-01, as shown in the curve of right y axis of Figure 8, the media request rate is almost a constant in the first three months, and then suddenly increases at the beginning of the fourth month due to a system upgrade, but still approximates to a constant.

The curve on the left y axis of Figure 8 shows the cumulative number of objects requested over time for workload ST-SVR-01. The figure shows that the number of objects requested in the workload increases quickly at the beginning, then the increase slows down and converges to a linear function asymptotically. This effect can be explained as follows. The initial non-linear increase of the cumulative number of requested objects is due to objects that were created and had been requested before the trace collection. However, the number of requests to these pre-existing objects decreases quickly with time, and the cumulative number of requested objects will be dominated by the objects born after trace collection after a certain duration of time. Thus, the linear part of the curve corresponds to the birth of new objects requested in this workload, indicating that the object birth rate is a constant in a coarse time granularity. We have similar observations on workloads BT-03 and IFILM-06.

The constant media request rates and object birth rates observed in these workloads indicate that these media systems evolve homogeneously over time during the trace collection period. We have analyzed the reference rank distributions of media objects accessed in different weeks and different number of weeks in these three workloads. *All of them can be well fitted with the stretched exponential distribution* (with $R^2 > 0.95$ for most durations and $R^2 > 0.93$ for all durations, also accepted by χ^2 tests). Furthermore, in each workload, we find that parameter c is a constant for different weeks and different number of weeks of that workload. According to its correlation with the median file size in the workload as presented in Section 4.1, the time-invariant property of stretch factor c indicates that the median file size of a workload is a constant along time, which is also confirmed by our validation.

Figure 9 shows the reference rank distributions of BT-03 in different number of weeks. We can see that the minus of the slope of the fitted line, i.e., the parameter a of the corresponding SE distribution, increases with time gradually. Figure 10 further shows the evolution of parameter a over time in these three workloads. This evolution corresponds to the observation in the study of Cherkasova et al. (with workloads HPC-98 and HPLabs-99) [14]: for monthly workloads, the access pattern looks like a Zipf-like distribution, while for workloads longer than 6 months, it does not. Actually, even for monthly workloads, the Zipf-like fitting of reference rank distribution is quite rough, as shown in the figures of [14]. In contrast, the media reference ranks of a workload in different durations can be well fitted with SE distributions with the same stretch factor c and a different parameter a.

Now let us consider a homogeneously evolving media system with a constant media request rate λ_{req} and a constant object birth rate λ_{obj}. In a coarse time granularity, this is a reasonable assumption for many systems, such as the three workloads discussed above. According to Equation 5, $\langle y_{se} \rangle$ increases with time since c is a constant and a increases with time (denoted as $a(t)$ in the follows).

The cumulative number of requested objects $N(t)$ in the time duration $[0, t)$ is

$$N(t) = \lambda_{obj} t + N'(t), \tag{6}$$

where $\lambda_{obj} t$ is the number of requested objects born in time $[0, t)$. $N'(t)$ is the number of "old" objects that are born before $t = 0$ and requested in time $[0, t)$. Denote the cumulative number of requests to object i in the workload as $y_i(t)$ $(1 \leq i \leq N(t))$. Denoting the mean value of the cumulative requests to objects in time $[0, t)$ as $\langle y(t) \rangle$, we have

$$\langle y(t) \rangle = \frac{\lambda_{req} t}{N(t)} = \frac{\lambda_{req}}{\lambda_{obj}} \frac{1}{1 + \frac{N'(t)}{\lambda_{obj} t}}. \tag{7}$$

Intuitively, $\lim\limits_{t \to \infty} \frac{N'(t)}{\lambda_{obj} t} = 0$, since the number of old objects will be requested less frequently with the passage of time. Assuming the popularity of a media object decreases exponentially with time [21], we have $N'(t) = O(\log t)$ (see Appendix D of [20] for a brief proof).

Apply Equation 4 on $\langle y(t) \rangle$ in Equation 7, we have

$$a(t) = \left[\frac{\langle y(t) \rangle}{\Gamma(1 + \frac{1}{c})} \right]^c = \left[\frac{1}{\Gamma(1 + \frac{1}{c})\big(1 + \frac{N'(t)}{\lambda_{obj} t}\big)} \frac{\lambda_{req}}{\lambda_{obj}} \right]^c. \tag{8}$$

Thus, if λ_{req}, λ_{obj}, and c are all constants, we have

$$\lim_{t \to \infty} a(t) = \left[\frac{1}{\Gamma(1 + \frac{1}{c})} \frac{\lambda_{req}}{\lambda_{obj}} \right]^c. \tag{9}$$

In all workloads, the stretch factor c is less than 2. When $0 < c \leq 2$, $\frac{1}{\Gamma(1+\frac{1}{c})}$ increases with c. Thus, for the reference rank distribution of a media workload, a increases with stretch factor c, the ratio of media request rate to object birth rate, $\frac{\lambda_{req}}{\lambda_{obj}}$, and the duration of workload collection time t. In Section 4.3, we will further show that the deviation of media reference rank distribution from the Zipf model increases with parameter c and the length of workload duration t.

As shown in Figure 10(a) and 10(b), the increase of a in workloads BT-03 and IFILM-06 slows down with time. For workload ST-SVR-01, as shown in 10(a), the increase of a slows down in the first three months, then speeds up at the beginning of the fourth month suddenly, and then slows down again. The sudden increase of a in Figure 10(a) is caused by the sudden increase of λ_{req} due to a system upgrade, which means a may not converge to a constant in practice.

For short duration workloads PS-CLT-04, ST-CLT-04, and ST-CLT-05, we compute the distribution parameters for workloads of different time intervals. We find both parameter c and parameter a are constant. The reason is as follows. For short term workloads, the number of requested objects is dominated by old objects. As shown in Figures 13 and 14, throughout the workload duration, $N'(t)$ increases with time almost linearly. According to Equation 8, this means the parameter a is a constant during this time. Thus, the popularity distribution is *stationary* for short duration workloads.

In summary, for a media system with homogeneous evolution, the media object reference rank distribution is non-stationary, which evolves over time with an increasing parameter a and a constant, time-invariant stretch factor c. This evolution is due to the effect of objects born before the workload collection and/or objects pre-existing in the system, which has significant impact on the caching performance of media systems.

4.3 Deviation of media access patterns from the Zipf model

In Section 3, we have shown that the distribution of media reference ranks has a fat head and a thin tail in log-log scale, deviating from the Zipf model. In order to quantitatively measure this deviation, Figure 11(a) shows a general stretched exponential distribution curve in log-log scale. In this figure, chord AB on the SE curve corresponds to the Zipf-like distribution (see Equation 14 of Section 5.1.1) with the same number of objects and the same number of references to the most popular object as those in the SE distribution. CD is parallel to AB and tangent to the SE curve at point (X_0, Y_0). If we use $|OE|$ to represent the distance from the original point to the chord AB and $|EF|$ to represent the distance between AB and CD, $|EF|/|OE|$ reflects the difference between the SE distribution and the corresponding Zipf-like distribution in log-log scale.

In Equation 2, let $X = \log i$, $Y = \log y_i$, we have

$$Y = \frac{1}{c} \log(b - aX), \tag{10}$$

where $0 \leq X \leq \log N$, $0 \leq Y \leq \frac{1}{c} \log b$.

Since the slope of chord AB is $k = -\frac{Y_{max}}{X_{max}} = -\frac{\log b}{c \log N}$, chord AB can be expressed as

$$Y = kX + Y_{max}. \tag{11}$$

Similarly, tangent CD can be expressed as

$$Y = kX + Y_0 - kX_0, \tag{12}$$

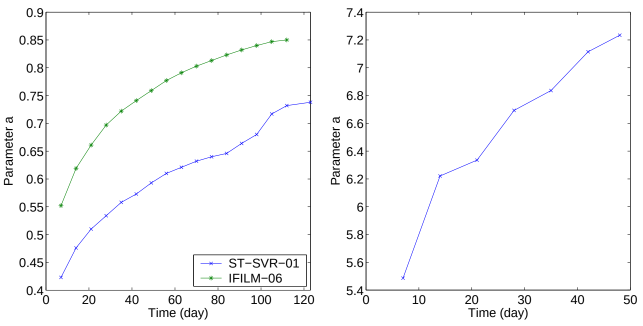
where $X_0 = \frac{1}{a}(b + \frac{a}{ck}), Y_0 = \frac{1}{c} \log(-\frac{a}{ck})$. Thus,

$$\begin{aligned} \frac{|EF|}{|OE|} &= \frac{|AC|}{|OA|} = \frac{Y_0 - kX_0 - Y_{max}}{Y_{max}} = \frac{Y_0 - kX_0}{Y_{max}} - 1 \\ &= \frac{\log(a \log N) - \log\log(1 + a \log N) - 1}{\log(1 + a \log N)} + \frac{1}{a \log N}, \end{aligned} \tag{13}$$

for which we have $\frac{|EF|}{|OE|} \to 1$ when $a \log N \to \infty$. Thus, in log-log scale, the difference between the media reference rank distribution and the Zipf model increases with $a \log N$. Figure 11(b) shows $\frac{|EF|}{|OE|}$ with different values of a and N.

For a homogeneously evolving media system, a evolves along time until it approaches to a constant (Equations 8 and 9), and N increases with time linearly (Equation 6). Thus, in log-log scale, the deviation of media reference rank distribution from the Zipf model increases with time, causing the "fat head" and the "thin tail" of the distribution curve.

We have compared this deviation for different periods of long duration workloads. For example, for the first week of workload ST-SVR-01, where $a = 0.423$ and $N = 459$, we have $\frac{|EF|}{|OE|} = 0.16$. In contrast, for the entire 4 months of workload ST-SVR-01, where $a = 0.738$ and $N = 2260$, we have $\frac{|EF|}{|OE|} =$



(a) ST-SVR-01, IFILM-06 (b) BT-03

Figure 10: Evolutions of reference rank distributions (parameter a in the SE model)

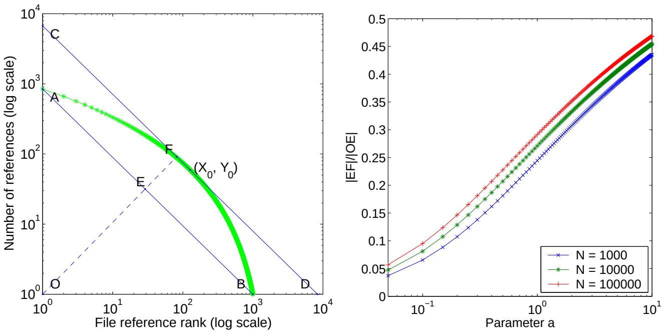
0.23. This evolution is consistent with the observation in the study of Cherkasova et al. [14] (with workloads HPC-98 and HPLabs-99), as we have presented in Section 4.2.

Since term $a(t)$ (Equation 8) and term $\log N(t)$ increase with time t slowly, it may take a long time to observe a significant deviation between the media popularity distribution and the Zipf model in log-log plot, especially for workloads with small media files, such as Web media workloads in Figure 1 [4]. *However, this does not mean that the Zipf model is a good approximation of media access patterns for workloads with short durations.* Since log scale compresses data distributions significantly, a small difference in the Zipf model may correspond to a significantly larger difference in the SE model. As shown in Figures 1(b) (PS-CLT-04), 1(c) (ST-CLT-04), and 1(d) (ST-CLT-05), the deviations of these short duration workloads (9–11 days) are still non-trivial, though not significant. For the evaluation of system related metrics, such as hit ratios of media caching, using Zipf model may cause significant errors. We will further study this issue in Section 5.1.

Even for short duration workloads, the deviation can still be significant. According to Equation 8, parameter a depends on the stretch factor c, ratio of $\frac{\lambda_{req}}{\lambda_{obj}}$, and workload duration time t. For workloads with large files, usually both the stretch factor c and the ratio of $\frac{\lambda_{req}}{\lambda_{obj}}$ are large. We have presented the correlation between stretch factor c and file sizes in Section 4.1. The latter is large because large media files, such as movies, tend to have a high request rate and a low production rate. For example, as shown in Figure 10, the parameter a of one-week workload of BT-03 (median file size 636 MB, $c = 0.52$, $\frac{\lambda_{req}}{\lambda_{obj}} = 141.1$) is even much larger than that of the entire duration (4 months) of workload ST-SVR-01 (median file size 15 MB, $c = 0.2$, $\frac{\lambda_{req}}{\lambda_{obj}} = 99.6$), and the difference between its reference rank distribution and the corresponding Zipf-like distribution is significant. For workloads with large files and long durations, the deviation is more significant. As shown in Figures 2(a), 2(b), 3(a), and 3(c), the log-log plots of media reference rank distributions of workloads mMoD-98 (median file size 125 MB), CTVoD-04 (median file size 300 MB), KaZaa-02 (median file size 300 MB), and BT-03 (median file size 636 MB), deviate from a straight line remarkably.

5. IMPLICATIONS OF STRETCHED EXPONENTIAL ACCESS PATTERNS

The Zipf-like reference rank distribution of Web objects and the widely deployed Web proxies have demonstrated the significance of access patterns on the performance of content delivery systems [12]. In this section, we explore the implication of stretched exponential access patterns on media caching, from the aspect of the asymptotic properties of workload size and cache size.



(a) Comparison of log-log plot (b) Effects of a and N

Figure 11: The differences between SE and Zipf-like plots

5.1 Caching implications of different reference rank models

As we have shown in Section 4.2, the media reference rank distribution is *stationary* for short duration workloads. If we assume requests in a media workload are *independent* with each other, the temporal locality in the workload can be analyzed with the reference rank model of objects.

5.1.1 Caching with the Zipf model

A Zipf-like reference rank distribution can be described as

$$y_i = \frac{A}{i^{\alpha}}, \tag{14}$$

where y_i is the number of references to the i-th popular object ($1 \le i \le N$, N is the total number of objects in the workload), α (the skewness factor) is a constant that characterizes the shape of the distribution, and A is a normalization factor. Thus we have

$$\log y_i = \log A - \alpha \log i, \tag{15}$$

which means the distribution function is a straight line in log-log scale. Assuming $y_N = 1$ when N is large enough (the N-th object gets only one access), we have $A = N^{\alpha}$.

When $\alpha < 1$ and $N \to \infty$, the mean value of a Zipf-like distribution is

$$\lim_{N\to\infty} \langle y_{zf} \rangle = \lim_{N\to\infty} \tfrac{1}{N} \textstyle\sum_{i=1}^{N} y_i = \lim_{N\to\infty} N^{\alpha-1} \sum_{i=1}^{N} i^{-\alpha} = \tfrac{1}{1-\alpha}. \tag{16}$$

Assume each object occupies one unit of storage volume and the cache size is ηN, where $\eta \le 1$ and is a constant. With an optimal cache replacement, when $\alpha < 1$, the number of total cache hits is

$$\sum_{i=1}^{\eta N} (y_i - 1) \approx \int_{1}^{\eta N} (\frac{N}{x})^{\alpha} dx - \eta N \approx (\frac{\eta^{1-\alpha}}{1-\alpha} - \eta)N. \tag{17}$$

Thus, the optimal hit ratio of a workload with Zipf-like distribution is [5]

$$H_{zf}^{opt}(\eta) = \lim_{N\to\infty} \frac{1}{N\langle y_{zf}\rangle} (\frac{\eta^{1-\alpha}}{1-\alpha} - \eta)N = \eta^{1-\alpha} - \eta(1-\alpha). \tag{18}$$

According to the above derivation, we can estimate the optimal hit ratio. For example, when $\alpha = 0.8$, caching 25% of the data can achieve a hit ratio of 0.7. This is consistent with the Web caching hit ratios reported in [12].

[4] Similarly, the YouTube video access pattern in study [17] is also close to Zipf. Please see Section 3.5.4 of [20] for a detailed explanation.

[5] When $\alpha \ge 1$, we have $H_{zf}^{opt}(\eta) = 1$.

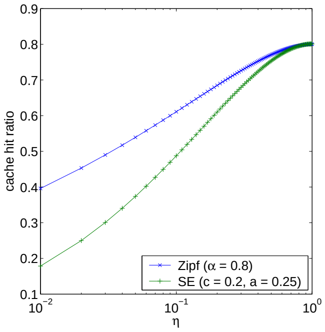


Figure 12: Caching efficiency

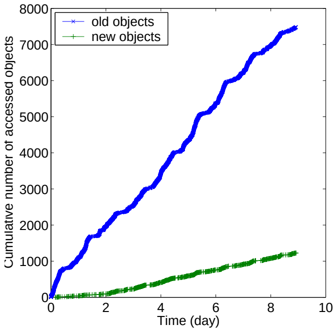


Figure 13: Requests in ST-CLT-04 (9 days)

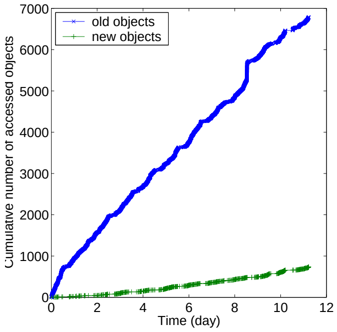


Figure 14: Requests in ST-CLT-05 (11 days)

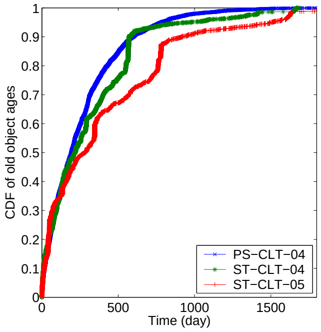


Figure 15: CDF of old object ages

5.1.2 Caching with the stretched exponential model

Assume each object occupies one unit of storage volume and the cache size is k ($k = \eta N$, $\eta \leq 1$ and is a constant). The optimal hit ratio of SE distribution is

$$\begin{aligned} H_{se}^{opt}(\eta) &= \lim_{N\to\infty} \frac{\text{hits}}{N\langle y_{se}\rangle} = \lim_{N\to\infty} \tfrac{1}{N\langle y_{se}\rangle} \textstyle\sum_{i=1}^{k}(y_i - 1) \\ &= \lim_{N\to\infty} [\tfrac{1}{\langle y_{se}\rangle} \textstyle\int_{\frac{1}{N}}^{\frac{k}{N}} (1 - a\log x)^{\frac{1}{c}} dx - \tfrac{k}{N\langle y_{se}\rangle}] \\ &= \frac{\Gamma(1+\frac{1}{c}) - \gamma(1+\frac{1}{c}, \frac{1}{a} - \log\eta)}{\Gamma(1+\frac{1}{c}) - \gamma(1+\frac{1}{c}, \frac{1}{a})} - \frac{\eta}{\langle y_{se}\rangle}. \end{aligned} \tag{19}$$

Figure 12 shows an exemplified comparison of optimal hit ratios between the Zipf-like and the stretched exponential models when η changes. The parameters are selected based on typical client side Web workloads ($\alpha = 0.8$) and media workloads ($c = 0.2, a = 0.25$, same as those in ST-CLT-05). Both distributions have the same hit ratio with an unlimited size of cache [6]. The duration of workload ST-CLT-05 is 11 days, comparable to those of client side Web workloads in Web caching studies such as Breslau et al. [12]. Thus, this comparison is fair. From Figure 12, it is clear that the caching efficiency of workloads under the SE model is much worse than that under the Zipf model: caching 1% Web content can achieve about 40% hit ratio, while caching 1% media content can only achieve 18% hit ratio.

In order to compare the asymptotic caching performance of these two models with a small η, we consider the case where the cache size k is a constant and $k << N$.

For a Zipf-like distribution with $\alpha < 1$, we have

$$H_{zf}^{opt}(\frac{k}{N}) = \frac{\sum_{i=1}^{k}(y_i - 1)}{\sum_{i=1}^{N} y_i} = \frac{\sum_{i=1}^{k}(\frac{N}{i})^{\alpha} - k}{N\langle y_{zf}\rangle} \approx \frac{\sum_{i=1}^{k}\frac{1}{i^{\alpha}}}{\langle y_{zf}\rangle}\frac{1}{N^{1-\alpha}}. \tag{20}$$

For a stretched exponential distribution, we have

$$\begin{aligned} H_{se}^{opt}(\tfrac{k}{N}) &= \frac{\sum_{i=1}^{k}(y_i - 1)}{\sum_{i=1}^{N} y_i} = \frac{\sum_{i=1}^{k}(1 + a\log\frac{N}{i})^{\frac{1}{c}} - k}{N\langle y_{se}\rangle} \\ &\leq \frac{k}{\langle y_{se}\rangle}\frac{(1 + a\log N)^{\frac{1}{c}}}{N}. \end{aligned} \tag{21}$$

When $N \to \infty$, we have

$$\lim_{N\to\infty}\frac{H_{se}^{opt}(\frac{k}{N})}{H_{zf}^{opt}(\frac{k}{N})} = \lim_{N\to\infty}\frac{\langle y_{zf}\rangle}{\langle y_{se}\rangle}\frac{k}{\sum_{i=1}^{k}\frac{1}{i^{\alpha}}}\frac{(1 + a\log N)^{\frac{1}{c}}}{N^{\alpha}} = 0. \tag{22}$$

Since $\langle y_{zf}\rangle$ and $\langle y_{se}\rangle$ are independent of k, this equation means the cache efficiency of a workload following the stretched exponential model is asymptotically lower than that of a workload following the Zipf model when η is small ($k << N$).

5.1.3 Simulation results of streaming media caching

Streaming media objects commonly have large file sizes and are often partially accessed. We have studied the efficiency of segment-based caching for streaming media through the reference rank distribution of media segments. We evenly divide each media object into segments of 5-second playback length, which is the default setting of the play-out buffer size of Windows media player [4]. The same analysis is also conducted with byte-based segmentation, which shows similar results.

Segment hit ratio reflects the actual byte hit ratio well since each segment is very small. We have studied the optimal hit ratio, as well as the hit ratio of different replacement algorithms on the streaming media workloads we collected. We find that the optimal segment hit ratio of streaming media is much lower than the corresponding optimal object hit ratio. For example, for workload ST-CLT-05, the maximal segment hit ratio (unlimited cache) is only 0.342, although the object hit ratio can be up to 0.6. Furthermore, the segment hit ratios of LRU and LFU replacement algorithms are much lower than the optimal segment hit ratio. For example, for workload ST-CLT-05, when the normalized cache size is 0.2, the optimal segment hit ratio is 0.342 while the LRU segment hit ratio is only 0.281. LFU achieves even worse performance, because it cannot capture the sequential access order of segments in an object as LRU. For more data and detailed analysis, please refer to Section 3.6.2 of [20].

5.1.4 Summary

The analysis above indicates that media caching under the stretched exponential model is far less effective than Web caching under the Zipf model, especially when only a small portion of accessed objects can be cached. Thus, in many previous studies advocating media caching where the Zipf model is assumed, it is highly likely that the benefit of media caching has been overestimated.

5.2 The evolution of media caching performance

As presented in Section 4.2, in a homogeneously evolving media system, due to the diminishing accesses to old objects, the average number of requests per object in a media workload gradually increases with time until approaching to a constant. With a higher a and a constant c over time, the request concentration of media requests increases, and the performance of media caching can be improved. Now we study how long it takes and how much storage is required for a significant performance improvement.

We have extracted the `Last-Modified` field in the RTSP/HTTP headers of media requests in workloads PS-CLT-04, ST-CLT-04, and ST-CLT-05. We use this field to approximate the birth time of a requested object. Figures 13 and 14 show the cumulative number of objects requested over time in ST-CLT-04 and ST-CLT-05, respectively. We plot ob-

[6] The actual hit ratio of ST-CLT-05 is only 0.6 for unlimited caching, because many objects are accessed only once.

jects born before workload collection (old objects) and object born after workload collection (new objects) separately. In all three workloads, most objects requested in the workload are old objects pre-existing before workload collection, and the numbers of old and new objects both increase with time linearly. Figure 15 shows the CDF of the ages of old objects when they were requested for the first time in the three workloads. We can see that more than 70% of old objects are at most 500 days old. This figure indicates that the number of old objects will not linearly increase with time forever, and new objects will catch up with and overwhelm old objects after a long time.

Assume all objects are cacheable and have the same file size. According to Equation 7, when new objects dominate the workload, the average number of requests to an object in workloads PS-CLT-04, ST-CLT-04, and ST-CLT-05 will be 39.7, 30.3, and 40.5, respectively, and the hit ratio for caching 10% objects will be 0.85, 0.84, 0.85, respectively. In contrast, in these workloads, the optimal hit ratios for caching 10% objects are only 0.52, 0.48, and 0.54, respectively.

However, this process may take a very long time. Even assuming the number of old objects will not increase after the workload collection (an overoptimistic assumption according the trends shown in Figures 13 and 14), the time for new objects to dominate the workload is 25.3, 54.4, and 103.3 days for PS-CLT-04, ST-CLT-04, and ST-CLT-05, respectively. For server side workloads such as ST-SVR-01 (Figure 8), this time can be shorter, because the number of objects in a server is limited. However, the duration of this time is still in weeks.

Considering that the popularity of a media object decreases with time, the request correlation, another source of temporal locality of media systems, could be exploited to further improve the caching performance. However, this process also takes time, since media objects have long life spans. For a short duration, it is hard to exploit correlation among requests.

Thus, for client side media caching, a huge storage space is required and the performance can hardly be improved without a long execution time. A distributed caching system with enormously large storage and huge amount of pre-existing content, such as a P2P-based streaming content delivery system, can be very effective. In a P2P system, the total amount of storage is scalable to the user population and workload size. Furthermore, the content cached in a client's local machine can be contributed to the system when the client joins the system for the first time, reducing the cold misses of new requests and thus significantly reducing the system execution time to achieve the optimal performance.

6. CONCLUSION

In this paper, we have analyzed a wide variety of media workloads collected from both the client side and the server side in different media systems with different delivery approaches, and proposed a general model for Internet media access patterns. We have found that the reference ranks of media objects in these workloads follow the stretched exponential distribution, whose shape is determined by the media file sizes and the aging effect of media accesses in the workload. Modeling the performance of media caching, we found that the performance of media caching under the stretched exponential distribution is far less efficient than Web caching under the Zipf-like distribution, indicating previous studies based on the Zipf assumption have potentially overestimated the benefit of media caching. An efficient media delivery system thus needs to leverage distributed resource sharing to scale its storage space with the increase of its workload size along time. Our study indicates that a client-server based caching system, such as a Web proxy, would not be able to effectively deliver media contents in a scalable manner, and provides an analytical basis for applying a P2P model to build large scale Internet media delivery systems.

Acknowledgments

We thank the anonymous referees for providing constructive comments. The work has been supported in part by the U. S. National Science Foundation under grants CNS-0509054, CNS-0509061, CNS-0621629, CNS-0621631, and CNS-0746649; and by a China MOST project (2006BAH02A10).

7. REFERENCES

[1] http://iblnews.com/story.php?id=17429.

[2] http://www.youtube.com/.

[3] http://www.imdb.com/chart/top.

[4] Buffer settings in windows media player. http://support.microsoft.com/.

[5] Helix universal proxy. http://www.realnetworks.com/.

[6] IFILM. http://www.ifilm.com/.

[7] Using the Microsoft Windows media proxy with ACNS 5.1. http://www.cisco.com/.

[8] Windows media services. http://www.microsoft.com/.

[9] S. Acharya, B. Smith, and P. Parnes. Characterizing user access to videos on the world wide web. In *Proc. of MMCN*, 2000.

[10] M. Arlitt and C. Williamson. Web server workload characterization: The search for invariants. In *Proc. of ACM SIGMETRICS*, May 1996.

[11] A. Bellissimo, B. Levine, and P. Shenoy. Exploring the use of BitTorrent as the basis for a large trace repository. Technical report, Department of Computer Science, University of Massachusetts, Amherst, 2004.

[12] L. Breslau, P. Cao, L. Fan, G. Philips, and S. Shenker. Web caching and Zipf-like distributions: Evidence and implications. In *Proc. of INFOCOM*, Mar. 1999.

[13] M. Cha, H. Kwak, P. Rodriguez, Y. Ahn, and S. Moon. I tube, you tube, everybody tubes: Analyzing the world's largest user generated content video system. In *Proc. of ACM SIGCOMM IMC*, Oct. 2007.

[14] L. Cherkasova and M. Gupta. Characterizing locality, evolution, and life span of accesses in enterprise media server workloads. In *Proc. of ACM NOSSDAV*, May 2002.

[15] M. Chesire, A. Wolman, G. Voelker, and H. Levy. Measurement and analysis of a streaming media workload. In *Proc. of USENIX USITS*, Mar. 2001.

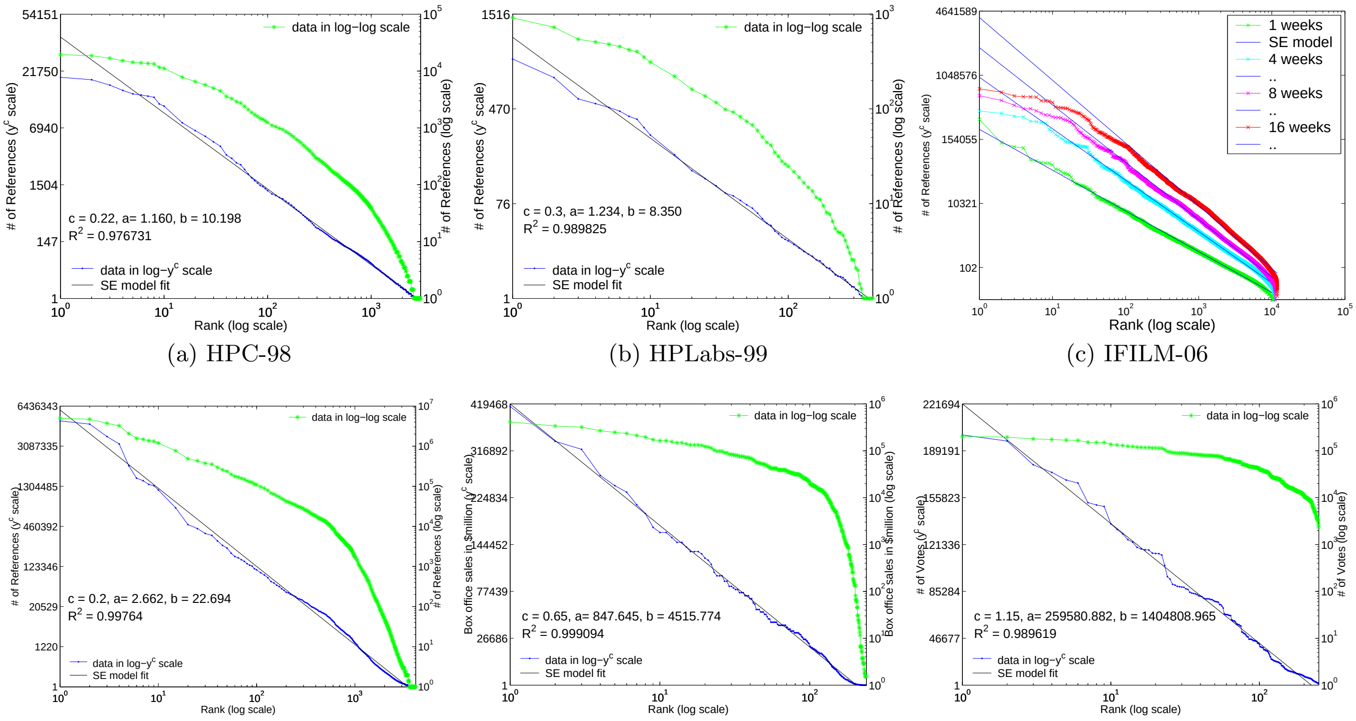
[16] J. Chu, K. Labonte, and B. Levine. Availability and popularity measurements of peer-to-peer file systems. In *Proc. of SPIE ITCom*, July 2002.

[17] P. Gill, M. Arlitt, Z. Li, and A. Mahanti. YouTube traffic characterization: A view from the edge. In *Proc. of ACM SIGCOMM IMC*, Oct. 2007.

[18] C. Griwodz, M. Bar, and L. Wolf. Long-term movie popularity models in video-on-demand systems or the life of an on-demand movie. In *Proc. of ACM Multimedia*, Nov. 1997.

[19] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan. Measurement, modeling, and analysis of a peer-to-peer file-sharing workload. In *Proc. of ACM SOSP*, Oct. 2003.

[20] L. Guo. *Insights into Access Patterns of Internet Media Systems: Measurements, Analysis, and System Design.* PhD thesis, The Ohio State University, Nov. 2007. http://www.cse.ohio-state.edu/~lguo/papers/thesis.pdf.



(a) HPC-98 (b) HPLabs-99 (c) IFILM-06

(d) Akamai-03 (e) Movie-02 (f) IMDB-06

Figure 16: More reference rank distributions of media workloads

[21] L. Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang. Measurements, analysis, and modeling of BitTorrent-like systems. In *Proc. of ACM SIGCOMM IMC*, Oct. 2005.

[22] L. Guo, S. Chen, Z. Xiao, and X. Zhang. DISC: Dynamic interleaved segment caching for interactive streaming. In *Proc. of IEEE ICDCS*, June 2005.

[23] L. Guo, E. Tan, S. Chen, Z. Xiao, O. Spatscheck, and X. Zhang. Delving into Internet streaming media delivery: A quality and resource utilization perspective. In *Proc. of ACM SIGCOMM IMC*, Oct. 2006.

[24] A. Iamnitchi, M. Ripeanu, and I. Foster. Small-world file-sharing communities. In *Proc. of IEEE INFOCOM*, Mar. 2004.

[25] J. Laherrere and D. Sornette. Stretched exponential distributions in nature and economy: "fat tails" with characteristic scales. *European Physical Journal B*, 2:525–539, 1998.

[26] O. Saleh and M. Hefeeda. Modeling and caching of peer-to-peer traffic. In *Proc. of IEEE ICNP*, Nov. 2006.

[27] K. Sripanidkulchai, B. Maggs, and H. Zhang. An analysis of live streaming workloads on the Internet. In *Proc. of ACM SIGCOMM IMC*, Oct. 2004.

[28] W. Tang, Y. Fu, L. Cherkasova, and A. Vahdat. MediSyn: A synthetic streaming media service workload generator. In *Proc. of ACM NOSSDAV*, June 2003.

[29] E. Veloso, V. Almeida, W. Meira, A. Bestavros, and S. Jin. A hierarchical characterization of a live streaming media workload. In *Proc. of ACM SIGCOMM IMW*, Nov. 2002.

[30] A. Williams, M. Arlitt, C. Williamson, and K. Barker. *Web Content Delivery*, chapter Web Workload Characterization: Ten Years Later. Springer, 2005.

[31] C. Williamson. On filter effects in Web caching hierarchies. *ACM Transactions on Internet Technology*, 2(1):47–77, 2002.

[32] H. Yu, D. Zheng, B. Y. Zhao, and W. Zheng. Understanding user behavior in large scale video-on-demand systems. In *Proc. of ACM EuroSys*, Apr. 2006.

APPENDIX

A. MORE REFERENCE RANK DISTRIBUTIONS OF MEDIA WORKLOADS

Figure 16(a) and 16(b) show the reference rank distribution of media objects in workloads HPC-98 and HPLabs-99, respectively. Figure 16(c) shows the reference rank distributions of workload IFILM-06 within different durations. Similar to Figure 9, the minus of the slope of the distribution curve increases with time. The head of the reference rank distribution (raw data) deviates from the SE model gradually due to the accumulation of caching effect, as presented in Section 3.2.2. Furthermore, the tail of the distribution (raw data) is "cut off" from the SE model with time gradually. This is because for a server side media workload, the media requests are constrained by the number of objects introduced in the system. If the object birth rate is small, even the least popular object can still get a certain amount of accesses, and thus Equation 3 should be changed to $b = y_N + a \log N$ where $y_N > 1$. However, this does not affect the analysis in the paper since b is only a normalization parameter.

Figure 16(d), 16(e) and 16(f) show the reference rank distributions in media workloads of the "other" category. As shown in Figure 16(d), the stretch factor of live streaming media workload Akamai-03 is 0.2, very close to that of most on-demand Web media workloads. Figure 16(e) and 16(f) show the reference rank distribution of media objects in workload Movie-02 (movie box office sales) and IMDB-06 (top 250 movie votes), respectively. The parameter a of these two workloads are much higher than other workloads, since they have small media birth rate and large media request rate.