基于分段流媒体代理 Cache 的策略研究*)

The Study of Strategy Based-Segment of Proxy Cache for Streaming Media

许志闻　庞云阶　王钲旋　郭晓新
(吉林大学计算机科学与技术学院　长春130021)

Abstract The proxy cache for streaming media is efficient method to solve network congestion. In this paper, we propose a strategy of cache-multicast based on partial cache strategy for the prefix cache/suffix cache and based-segment cache. This strategy is that the multicast video for the same time replaced by the use of interval time. Making advantage of that reference frequency of interval time is high than reference frequency of whole media, it enhances the efficiency for streaming media requested by many users, it saves the traffic resource for network backbone. Event-driven simulations are used to evaluate this cache-multicast approach. The results show that: (1) Cache-multicast strategy is effective not only in increasing byte-hit ratio and reducing traffic of network backbone, but also in lowering the number of requests that require delayed starts. (2) Cache-multicast strategy is especially advantageous when the cache size is limited, the set of hot media objects changes over time, many users play the same media in a interval time, the media file size is large, or when many users may stop playing the media after only a few initial blocks.

Keywords Streaming media, Proxy cache, Proxy cache for streaming media, Cache-multicast

1. 引言

WWW 的爆炸性增长已经使在 Internet 应用中的用户网络拥塞现象明显增加，一种通用的减少网络拥塞的方法是在靠近使用 Internet 用户端使用代理 Cache，代理 Cache 保存着最近客户请求的 Web 对象，在没有与内容服务器相连时，去满足以后的客户对保存成 Web 对象的请求，客户不用再去内容服务器申请 Web 对象，从而减少反应时间和网络交通费用。目前基于 Web 的代理 Cache 得到了广泛的应用，但缓存文本和图像对象的代理 Cache 技术不适合于缓存流媒体，主要原因是：首先，视频文件特别大，一个单个文件可能要求10M 到10G 的存储空间，这依据于视频的质量和长度；这样大多数的缓存内容一定要存在磁盘上，而且磁盘缓存和主内存缓存一定要仔细地组织。其次实时媒体文件传输需要明显大的磁盘空间和网络带宽，要长时间内支持，这就要求在当前的 Cache 内容中，采用有效的缓存策略，避免使用太多的磁盘空间去保存 Cache 中的新内容。第三，客户可能仅对文件的播放部分感兴趣，这时他们请求最感兴趣的部分视频，这说明部分文件是重要的。由于流媒体的特点决定它不象代理 Cache 那样只是缓存 Web 对象，而是缓存流媒体对象，这个特点使流媒体代理 Cache 技术成为挑战性的研究工作。

2. 相关工作

2.1 前缀 Cache/后缀 Cache 策略

流媒体代理 Cache 还处在理论研究和实验研制的发展阶段，为了解决媒体的开始反应时间和数据传送的平稳，Z. Miao 等人提出了前缀 Cache/后缀 Cache 的策略[1,3]，在传送的过程中把媒体分成两部分传送，前面的部分比较小，叫前缀 Cache，后面的部分叫后缀 Cache。前缀 Cache 一般保存在代理 Cache 中，当客户申请时，先播放前缀 Cache 部分媒体，并把后缀 Cache 的部分从内容服务器传送到代理服务器中，当播放完前缀 Cache 部分媒体，再播放后缀 Cache 的部分。这种策略有效地解决了开始延迟时间问题。

2.2 基于分段 Cache 策略

从代理 Cache 的角度出发，媒体流的开始部分比后面的部分更加重要。Wu. Kun-Lun 等人根据大多数的媒体对象开始部分的重要性和大多数媒体对象应该被部分缓存的调查，开发了基于分段的策略去代理高速缓存大多数媒体对象[2]，代理 Cache 缓存的媒体对象段根据体积变化、距离变化分成段。事实上，分段的大小是指数增加；简单地说，段 i 的大小是2^{i-1}个块，它包含媒体块编号为2^{i-1}，$2^{i-1}+1$，…，2^{i-1}(其中 i=1,2,…,M)(图1)。指数大小变化段的形成动机是为了可以快速放弃大的缓存媒体对象块，该 Cache 管理方法可以快速地调整部分缓存对象。在单一媒体作用中，Cache 管理者可以释放缓存媒体对象的1/2。对于每一个对象缓存的段数由缓存的准入控制和替换算法动态确定，它们携带着以参考频率和从媒体开始到段的距离为基础的不同段的缓存值，缓存策略给出了对开始段的优先处理，因

*)此课题得到了国家教委博士点基金项目(编号：20010183041)资助。

为初始段决定着被用户察觉的延迟反应；许多观察者可能在仅观看初始段后，就决定停止观看，在这种情况下甚至不需要把后面的段预先取来，后面的段因此要逐渐变大，由此可以减少代理服务需要跟踪和管理段的数目。

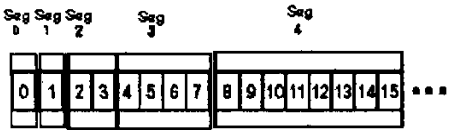

图1 分段方法

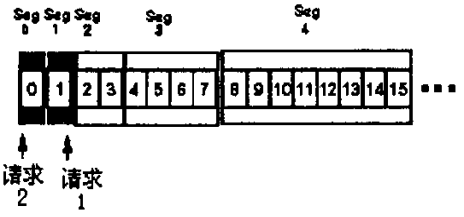

图2 缓存-组播状态

2.3 基于分段 Cache 的准入控制策略和替换策略

考虑两个重要的缓存组织策略，一个是缓存的准入控制策略，另一个是缓存替换策略，这两个是紧密相关的。Cache 准入控制根据媒体普及性(population)判断是否允许缓存媒体对象段，准入控制对于相同媒体对象的不同段应用不同的标准。替换策略根据媒体的参考频率采用 LRU 替换，保证代理 Cache 具有较高的效率。

3. 分段代理 Cache 的缓存-组播策略

3.1 缓存-组播策略

组播也叫多点传送，它是网络传输中提高效率的有效方法。视频组播是指按照网络交换机和路由器逐级传送视频，形成的组播；利用代理 Cache 可以方便地实现组播，服务器-代理 Cache 之间是单播，代理 Cache-客户是多播，代理 Cache 代替了网络交换机完成组播传送，它要求所有的用户必须在同一时间接收和传送相同视频内容，这限制了视频组播的应用。在基于分段流媒体代理服务器中，缓存媒体的大小是随被申请播放媒体的频率变化的；通常情况下，一部分被保存在代理服务器中，另一部分不保存在代理服务器中，当客户申请这个媒体时，未保存在代理服务器中的媒体部分要从内容服务器中取出，传送到代理服务器之后，再传送给客户。当多个客户申请这个媒体时，就需要从内容服务器中多次传送媒体到代理服务器。我们利用流媒体代理 Cache 的特点，提出了缓存-组播策略，在一段时间里，如果存在两个或两个以上的客户申请播放同一个视频素材时，用缓存动态地保存一段视频，满足这段时间中多用户的申请需要，代理服务器不需要多次向内容服务器申请视频素材，只需要申请一次即可满足多个客户的申请需要。在这里，这段时间小于视频素材的时间长度，等于视频已被缓存的长度，且这段时间内它的参考频率大于准许缓存的参考频率。

在变体分段策略中[2]，根据媒体被用户申请的频率，代理 Cache 缓存不同的段，这种策略充分考虑了被申请媒体开始的重要性，保证代理 Cache 具有较高的效率。但基于分段的策略没有充分考虑未被保存在代理 Cache 的媒体部分对它的效率的影响，往往有很多用户会在相近的一段时间内观看同一个媒体，在这段时间内，媒体的参考频率会很高，我们设计的组播-缓存策略的主要思想是在分段策略的基础上，充分考虑相近一段时间内多用户申请观看同一个媒体的情况，用一段缓存动态跟踪多用户请求的媒体，保证多用户只有第一个用户需要从内容服务器取出未缓存在代理 Cache 中的媒体部分，并把这段媒体缓存在代理 Cache 中一段时间，用这段时间长度作为缓存媒体的长度，并采用 FIFO 替换使缓存的媒体部分一直满足这段时间内多用户的需要。这段时间长度的确定会影响代理 Cache 的效率，我们可以通过对用户行为的计算找出这个时间长度，代理 Cache 根据这个长度分配缓存长度，这会给代理 Cache 的管理带来一定的麻烦，代理 Cache 的准入控制和替换策略会变得较复杂，根据用户的行为计算效率最高的时间间隔，也会给代理 Cache 带来负担。我们采用了一种简单、实用、高效的方法，采用代理 Cache 中已缓存的媒体长度作为缓存-组播的缓存长度，这样确定其代理 Cache 的准入控制和替换策略可以采用同一标准，简化了代理 Cache 的管理工作；这样确定缓存的长度，保证了高普及性的媒体用于缓存-组播的缓存长度大，较充分提高代理 Cache 的效率。变体积分段策略考虑开始段对代理 Cache 效率的影响，缓存-组播策略在考虑开始段的基础上，考虑了相近一段时间内多用户请求同一媒体对代理 Cache 效率的影响，进一步提高了代理 Cache 的效率。

3.2 缓存-组播策略的实现

具体实现缓存-组播策略如下：

A. 按图1基于变体积分段缓存策略分段，把内容服务器中被申请播放媒体分成相等的块，按2的指数倍递增分成段，0 Seg 作为缓存媒体的初始段 Kmin.，1 Seg 和0 Seg 相等，2 Seg 是1 Seg 的2倍，如此继续。

B. 代理 Cache 的缓存空间划分成两部分，一部分用于要缓存媒体的初始化段 Kmin，对每个被申

请媒体的初始化段 Kmin 采用 LRU 替换；另一部分缓存用于缓存初始化段后面的段和缓存-组播段，后面的段采用 LRU 替换，缓存-组播段采用 FIFO 替换，具体操作根据代理缓存的准入控制策略和替换策略执行。

C. 代理 Cache 按基于分段策略工作，对初始段和后面的变体积段采用 LRU 替换；这时媒体文件部分被保存在代理服务器中，在保存媒体的时间长度内如果有两个客户申请播放这个媒体文件产生组播，判断缓存-组播段时间内的参考频率是否大于或等于被缓存媒体对象的参考频率，如果不满足这个条件，取消缓存-组播的操作；如果满足，我们采用 FIFO 算法动态保存缓存-组播的那段文件内容，把代理服务器中已经缓存媒体文件的长度作为缓存-组播的长度，这个长度等于媒体分段的下一段的长度，从未被保存在代理服务器的文件段开始保存缓存-组播媒体文件内容。

D. 当流媒体代理缓存中保存的缓存-组播媒体内容正好达到它的长度时，判断代理 Cache 是否要进行变体积段的 LRU 替换，如果代理缓存随着客户申请观看这个媒体的频率增加需要增加变体积缓存段时，就把缓存在缓存-组播中的内容转换为代理缓存的下一变体积段保存起来，缓存-组播内容清空，又开始从未缓存媒体的位置缓存缓存一组播段，其缓存-组播段的长度增加一倍；否则不进行变体段的替换，保持缓存-组播的长度，采用 FIFO 算法动态地保存这段素材。

E. 缓存-组播这段素材至少满足两个以上客户的申请需要，当申请者中途停止申请减少到1时，停止缓存-组播段的 FIFO 动态缓存，记录缓存-组播媒体的位置 P，保留缓存-组播段中申请者申请媒体的位置到 P 段的内容，继续满足申请者的申请，直到位置 P，释放缓存并从这个位置继续调用内容服务器的媒体内容。

4. 性能评估

4.1 方法

我们应用一个事件驱动仿真器，模拟代理 Cache 服务去评估缓存-组播策略。在 Cache 中两个 LRU 和一个 FIFO 堆栈用于跟踪媒体对象；一个 LRU 堆栈是跟踪初始段而另一个 LRU 堆栈跟踪后面的段，FIFO 堆栈跟踪缓存-组播，总 Cache 能力的 C_{init} 部分说明初始段，后面的段在另一个 LRU 堆栈中被管理，但是基于分段 Cache 准许进入和替换策略被使用，而不使用简单 LRU；C_{multt} 用于描述缓存-组播策略，计算两个重要性能指标：字节命中率和带延迟开始的请求时间；字节命中率测量缓存对象的总字节和请求对象总字节的比率；当请求到达而且初始段 K_{min} 还未缓存在代理 Cache 中，它有开始延迟。

被请求的视频题目从特有的视频题目 M 的总和中选出，每一个视频题目M的普及性(population)跟随着 Zipf-like 分布 Zipf(x,M)[4]。Zipf-like 分布带来了两个参数，x 和 M，前面的与变形度相关，对于每个 $i\in\{1,\cdots,M\}$，分布由 $p_i=C/i^{1-x}$ 给出，这里 $C=1/\sum_{i=1}^{M}1/i^{1-x}$ 是一个标准化常量，设 X=0 时符合纯 Zipf 分布，它是高可变形的。在其它方面，设 X=1符合不带变形的统一分布，X 的缺省值是 0.2，这样对于 M 是2000。每个视频题目的普及性随时间改变，这常常用于模拟方案，在不同的时间里用户访问视频题目是不同的，而且他们的兴趣可能是不同的，在我们的仿真中，普及性分布改变每个请求 R；当它发生时，另一个与带有相同参数与 Zipf-like 较好相关分布被使用。

4.2 仿真结果

我们从 Cache 的大小、视频普及性变形、用户观察行为以及其它一些相关参数等方面，比较缓存-组播、全部视频、变体积分段及前缀/后缀策略的字节命中率和延迟开始方面的影响。

4.2.1 Cache 大小的影响

研究 Cache 大小在字节命中率和延迟开始的影响，全视频方法和前缀/后缀有相近的字节命中率(图3)，全视频策略和前缀/后缀有相近的字节命中率，在变体积段方法中的字节命中率对于较小的 Cache 大小是有明显的优点，缓存组播策略具有最好的字节命中率。因为相同数量的缓存能力用于初始块的存储，对于整个 Cache 范围内，缓存-组播、变体积分段和前缀/后缀有相同延迟开始。

4.2.2 视频普及性变形的影响

检查视频普及性变形对字节命中率和延迟开始方面的影响，在视频普及性的较宽范围变化中，缓存-组播策略带有最高的字节命中率；缓存-组播、变体积分段、前缀/后缀有相同的开始延迟的最小请求时间，都优于整个视频；对于 Zipf 参数 x，我们研究了普及性分布变化时，最大视频切换位置 k 的影响，图5显示视频最大切换位置的影响。

4.2.3 其它系统参数的影响

图6显示了视频长度对字节命中率的影响，缓存-组播策略和变体积段策略在大的媒体流代理 Cache 中特别有用，缓存-组播策略的优势更大。除了媒体文件的体积，特殊的媒体对象也影响着 Cache 的效率，通常这里有许多媒体对象存在于 Web 中，随着用户请求特殊对象的扩大，Cache 有效性降低。图7显示了可以被请求对象用户的情况，缓存-组播策略优点最突出。图8检验了用于存贮初始化段或前缀时缓存能力贡献的百分比，因为减少了

后面段或后缀的缓存能力，字节命中率随着使用初始段的增加而降低，在字节命中率中这种细小的降低可以通过减少延迟开始的实质性增加收益相抵消。

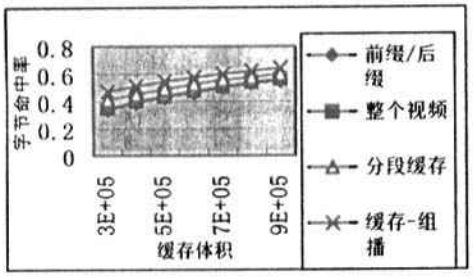

图3 缓存体积对字节命中率的影响

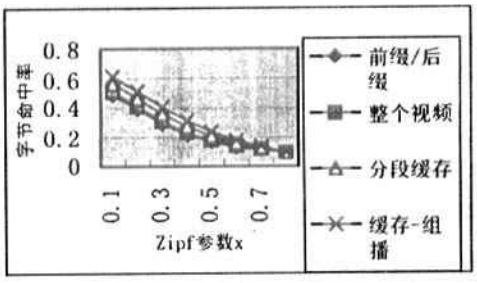

图4 视频普及性对字节命中率的影响

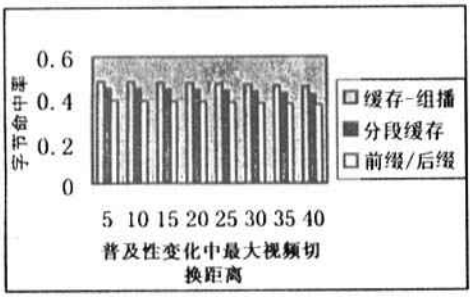

图5 视频最大切换距离的影响

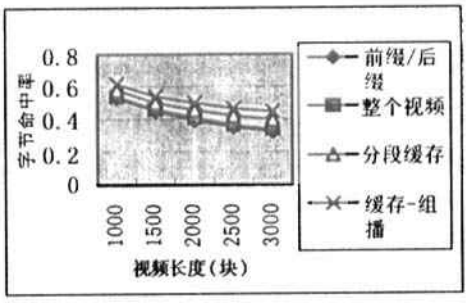

图6 视频长度的影响

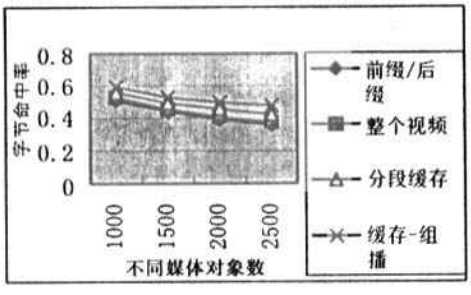

图7 不同媒体对象总数的影响

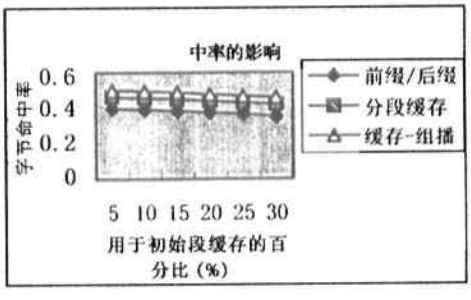

图8 用于初始段缓存的百分比对字节命中率的影响

4.2.4 用户观察行为的影响

用户观察行为影响着采用缓存-组播策略代理Cache的效率。用户观察媒体对象明显地影响着缓存-组播节省主干网络交通资源的程度，在已缓存的媒体对象的时间长度内，申请观看这个媒体对象越多，节省主干网络交通资源越多。研究在Web上的不完全相同用户行为的影响，大多数的用户可能在仅看第一段的一小部分后就停止播放视频，这里有许多关于用户过早地停止的原因。关于缓存-组播策略和变体积分段策略的交通比率是随过早停止观看视频的用户增加而降低的，因为我们总是预取一段，但部分复杂性影响缓存-组播方式对交通比率影响的程度，随着中途结束者的增多，缓存-组播方式降低其改进交通比率的幅度。

结论 我们提出缓存-组播策略，去代理高速缓存视频这样大的媒体对象，在Internet下作为流式视频和音频适当的代理Cache是非常重要的，甚至是很普及的；用简单的分段方法把媒体块分组成可变体积的段，不象Web对象那样处理整个视频，Cache的允入控制和替换策略把不同的缓存值分别指定不同的段，采用参考频率和从媒体开始到段的距离计算，这种缓存策略给出了初始段的参考处理，引导部分高速缓存对象从开始段开始缓存；缓存-组播利用变体积分段，充分考虑了用户的请求行为，在保证变体积分段的前提下，把多用户在一段时间内请求同一媒体的后部分用动态的缓存保存起来，保证用户的申请需要，在这段时间内参考频率高于该媒体的参考频率，缓存-组播策略有效地节省了主干网络的交通资源，进一步提高了代理缓存的字节命

（下转第86页）

表2 64:1恢复图像的 PSNR 值

	GHM	Cardbal2	Cardbal3	Cardbal4	单小波 D4
Lena	29.1278	29.3007	29.1474	29.3949	28.8061
Barbara	24.0355	24.0633	23.9534	24.1659	23.7683
Golhill	27.1109	27.3404	27.1844	27.4026	27.2066

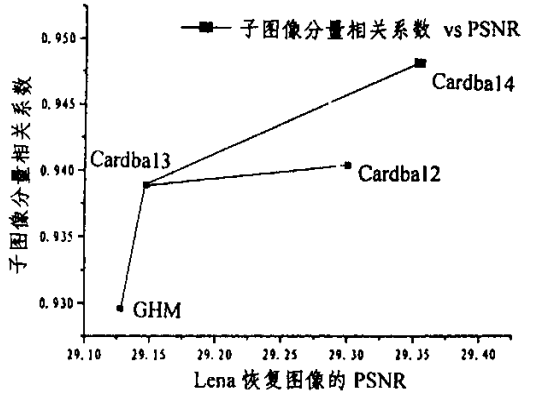

图2

近。于是，我们从大到小依次取$\frac{N\times N}{64}$个系数，称它们为重要系数，而后面的系数全部都认为不重要，全部置“0”。最后，由次链表中记录的位置信息，将重要系数置于原来位置，对于不重要系数的位置设为0。以上分析的所有多小波，都保留$\frac{1}{64}$的系数，重构以后得到的 PSNR 值。用这种方法我们可以不用做复杂的图像编码，同时又能够检测出哪一个多小波基适合我们即将处理的图像，能够实现压缩效果，从恢复的图像中我们可以算出 PSNR 值，展开对算法之间的比较。这样，我们可以对很多的多小波基做比较选择，参考以上的对称性，消失矩，逼近阶，能量集中特性，相关系数这些标准，最佳多小波基是 Carbal4，并且这类插值平衡多小波不必做预处理和后处理，能节约编码时间。图3图2给出了 PSNR 与能量集中特性，及相关系数之间的关系，我们可以直观地看出 Cardbal4这种多小波的优势。表2给出不同图像的仿真结果，可以看出多小波在处理高频含量较多的图像(如 Barbara)时，比单小波更有优势。

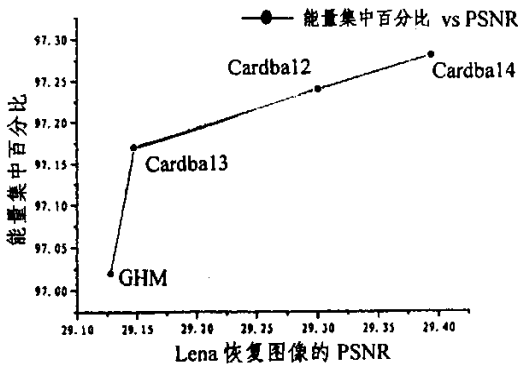

图3

结论 实际应用中，多小波的优势并不比单小波明显，但是多小波有很大的自由度和发展潜力，多个尺度函数决定了多小波比单小波具有更好的性质。多小波各个尺度函数之间的相关性，反映在图像中就是分解后分量图像的相关性，我们提出的新的多小波选择标准就是对其相关性的应用的一种探索，不仅有利于人们选择最佳多小波，而且有利于根据分量图像相关性、差异性进行多小波编码的研究。

参 考 文 献

1 Geronimo J S, Hardin D P, Massopust P R. Fractal functions and wavelet expansions based on several functions. J. Approx. Theory, 1994, 78(4): 373～401

2 Strela V, Walden A T. Orthogonal and biorthogonal multiwavelets for signal de-noising and image compression. SPIE Proc. 3391 AeroSense 98, Orlando, Florida, April 1998

3 Selesnick I W. Cardinal Multiwavelets and the Sampling Theorem. ICASSP-99, SPTM 3.3, 1999, 3: 1209～1212

4 Chui C K, Lian J A. A study of orthonormal multiwavelets. Applied Numerical Mathematics, 1996, 20(2): 273～298

5 Martin M B. Applciations of Multiwavelets to Image Compression. Thesis of Master of Science in E. E., Virginia Tech 1999

6 Selesnick I W. Balanced Multiwavelet Bases Based on Symmetric FIR Filters. IEEE Transactions on Signal Processing, 2000, 48: 184～191

(上接第71页)

中率和效率。我们用事件驱动仿真评估，从 Cache 的大小、视频普及性变形、用户观察行为以及其它一些相关参数对缓存-组播策略、变体积段策略、整个视频策略和一个前缀/后缀 Cache 策略进行了比较，结果表明：(1)缓存-组播策略不仅有效地降低了整个交通资源，增加了字节命中率，而且降低了请求开始的请求时间。(2)当 Cache 大小有限、用户请求集中在一段时间内、设置的热媒体(hot media)对象随时间改变、大量的媒体对象下展开请求、媒体文件体积很大，以及在许多用户仅看了一点初始段就停止观看视频时，缓存-组播策略都具有最佳的效果，它是提高流媒体代理 cache 效率的有效策略。

参 考 文 献

1 Miao Z, Ortega A. Proxy caching for efficient video services over the Internet. In: Proc. of Int, Web Caching Workshop, Apr. 1999

2 Wu K L, Yu P S. Segment-Based Proxy Caching of Multimedia Streams. In: Proc. Of IEEE INFOCOM, May 2001

3 Sen S, Reforrd J, Towsley D. Proxy prefix caching for multimedia streaming. In: Proc. of IEEE INFOCOM, Mar. 1999

4 Zipf G K. Hurnan Behaviour and the Principles of Least Effort, Addison-Wesley. Cambridge, MA, 1949