Optimal Pricing of Content Delivery Network (CDN) Services

Kartik Hosanagar, Ramayya Krishnan, Michael Smith, John Chuang
University of Pennsylvania, Carnegie Mellon University, Carnegie Mellon University,
University of California, Berkeley
{Contact: kartikh@andrew.cmu.edu}

Abstract

Content Delivery Networks (CDNs) intelligently cache content on behalf of content providers and deliver this content to end users. New services have been rolled out recently by CDNs that enable content providers to deliver entire web sites from the distributed CDN servers. Using analytical models, we address the optimal pricing of these services.

Our results suggest that, consistent with industry practices, CDN pricing functions should provide volume discounts to content providers. They also show that the most likely subscribers to CDN services are those content providers with high volume of traffic and with content having low security requirements. Significantly, our model also shows that larger CDN networks can charge higher prices in equilibrium, which should strengthen any technology-based economies of scale and make it more difficult for entrants to compete against incumbent firms. We find that CDNs will have to lower prices in light of increasing security concerns associated with content delivery. Alternatively, they will need to invest in developing and deploying technology to alleviate the security concerns. Finally, we find that declining bandwidth costs will negatively impact CDN revenues and profits.

1 Introduction

A Content Delivery Network (CDN) is a network of servers that cache or store web content (i.e., web pages) and intelligently deliver it to users based on their geographic location. When users request content, the request is redirected to the nearest CDN server. These CDN servers are typically collocated with ISPs with which the CDN has alliances. By delivering content from the edge of the Internet, CDNs serve to speed content delivery, circumvent bottlenecks and provide protection from sudden traffic surges that bring down servers and render web sites unreachable.

CDNs are an important element of the digital supply chain for delivery of information goods. The supply chain consists of Content Providers (CPs) that create the content; backbone and access networks that help transport the content; and CDNs that store and deliver the content to the end users. CDNs thus function as content storage and distribution centers performing similar functions as distributor/retailer warehouses in traditional supply chains. CDNs have become crucial to the content delivery strategy of web sites with high traffic. The key players in the CDN market include Akamai, Cable & Wireless, Speedera and Mirror Image. IDC projects that the CDN market will be worth over $2 billion in US alone by 2006.

In the recent years, CDNs have rolled out new services (for e.g., Akamai's Edgesuite[1]) that enable delivery of entire websites from the edge servers of CDNs. Conversations with executives from CDN firms (Maggs 2002) suggest that marketing managers find it challenging to determine the optimal pricing of these services. In this paper, we study optimal pricing of CDN services in a monopoly setting. We then analyze how recent trends such as declining bandwidth costs impact CDN pricing policies.

2 Value Proposition of CDNs

CDNs are a network of servers that store and deliver content on behalf of origin servers (and are thus also known as surrogate servers). CDN servers are geographically distributed at various locations at the edge of the network (typically collocated at partner ISP locations). Content of subscribing Content Providers (CPs) is stored in a number of these surrogate servers. When a request for content is received, the CDN delivers the content from the "nearest" server where nearness is based on expected latency, which is in turn determined by geographical proximity, server load and network conditions.

Client requests are redirected to CDN servers using either URL rewriting or DNS-based redirection (Krishnamurthy et al. 2001). With URL rewriting, the origin server rewrites URL links with CDN server addresses so that any click-throughs are directed to the CDN server. With DNS redirection, the CDN controls the

[1] http://www.akamai.com/en/html/services/edgesuite.html

DNS server of the CP and resolves the name to the IP address of a CDN server. The Time-To-Live (TTL) of these DNS mappings are typically kept small so that the CDN can map any given URL to different servers based on network conditions. CPs can selectively deliver some content from their servers and other content from CDN servers.

Krishnamurthy et al. (2001) found that 31% of a list of 127 popular websites used CDNs in 2000. This figure was up from 1-2% in 1999. Akamai, the largest CDN, is reported to have 80% of the CDN market share.

The primary benefit that a CDN provides a Content Provider (CP) is in scaling content delivery. The volume of traffic on the Internet is constantly increasing. Because of this, CPs must periodically resize and upgrade their server farms and bandwidth capacities to meet the traffic growth. This is exacerbated by the shift towards rich content such as multimedia. In addition, requests on the Internet have high variability. This creates challenging problems for CPs with regard to capacity allocation. If they allocate capacity based on peak traffic, their capacity will lie idle most of the time. Server farms and bandwidth are expensive enough to warrant optimizing on these parameters. If they under-provision, then the performance and uptimes of their web sites decrease significantly resulting in significant decrease in revenue. Because of these trade-offs, CPs have had to choose intermediate capacity levels and accept occasional down times and poor performance as a necessary evil. For example, flash crowds on Sep 11 overwhelmed media sites such as CNN and MSNBC and reduced site availability to close to 0% and increased response times to nearly 40 seconds when the sites were available[2]. Similarly, a flash crowd during Victoria's Secret Internet fashion show also resulted in their web server crashing.

CDNs provide CPs with a viable alternative to scale content delivery. CDNs maintain a network of servers around the world. Because of a CDN's ability to aggregate traffic across various sites, there are economies of scale in terms of infrastructure costs. Furthermore, the aggregation serves to reduce the impact of variability in demand for content[3]. This aggregation serves to provide sufficient buffers in capacities to handle flash crowd effects. Furthermore, since there are several nodes or servers from which the content can be served, no single point will be a bottleneck. Replication of content also helps to improve the availability of content, especially during Denial of Service (DoS) attacks.

Recently, CDNs have rolled out services that enable CPs to deliver entire websites from the edge servers. A well known example of such a service is Akamai's Edgesuite. The pricing of CDN services face different considerations than those faced by intermediaries in traditional supply chains. Another noticeable development in the industry has been a sharp decline in bandwidth costs, facilitated by improvements in technology and increased competition. This affects the cost structures of both CPs and CDNs and thus impacts pricing strategies. These questions – *how should CDN services such as Edgesuite be priced and how do external factors such as bandwidth costs and security considerations impact these pricing strategies* – are the focus of this paper.

3 Literature Review

CDNs have been widely studied in the computer science literature. Nottingham (2000) discusses the development of a framework to formally define the role of surrogate origin servers such as CDNs. Dilley et al. (2002) provide an overview of Akamai's network infrastructure and the technical challenges involved in operating a CDN. Gadde et al. (2000) explore the effectiveness of CDNs in the presence of conventional web proxy caching. Krishnamurthy et al. (2001) verify that CDNs reduce average download times but find that DNS redirection adds additional overheads. Johnson et al. (2000) also find that CDNs provide improvements in latency but find that they do not always choose the optimal server from which to serve the content. Chen et al. (2002) propose a protocol for dynamic placement of replicas in a CDN.

While the focus has generally been on the design of efficient CDN architectures, we feel that Information Systems (IS) research can make significant contributions to the study of CDNs. For example, Datta et al. (2002) motivate the importance of research on pricing of CDNs. Furthermore, managers in CDN firms face challenges in determining pricing schemes for the new services and need to take technological factors and Internet traffic patterns into account as they are a major determinant of these pricing strategies.

While pricing of traditional Telecommunications services has been studied in the past (for e.g., Mendelson and Whang (1990), Cocchi et al. (1993)), pricing of content delivery services is a relatively new and unexplored area. Hosanagar et al. (2002) have studied the optimal pricing of priority-based web proxy caching services. CDN pricing faces unique considerations such as bandwidth costs, traffic variability, security implications of outsourcing content delivery, etc. Our paper attempts to fill this void in the study of pricing strategies of CDNs.

4 Model

The model we present in this section assumes a monopoly CDN. Akamai, the leading CDN, has

[2] See http://news.com.com/2100-1023-272873.html. Retrieved March 2003.

[3] The law of large numbers and principles of statistical multiplexing imply that that capacity will be better utilized and overall impact of variability will reduce.

approximately 80% of the CDN market share and thus may be treated as a near monopoly. Consider a CP indexed by *i* delivering content to users. Let X be the random variable that denotes the number of requests to a CP in a period[4]. The CP does not know X a priori but knows the distribution *f(X)*. If the publisher sets up infrastructure to be able to process I requests per unit time on average (web server vendors typically specify the average request/minute their servers can process), then the CP's surplus from delivering content is $U_{self}(X) = V(X) - C(I) - c \cdot L(I,X)$. The net expected surplus from delivering content is $U_{self} = E[U_{self}] = V - C(I) - c \cdot L(I)$.

V(X) is the CP's benefit from responding to X requests. This could include revenues from selling products on the Internet, surplus from disseminating information, etc. In other words, *V(X)* is the reason why the CP maintains a website. The CP does not know X in any time period but knows the expected value denoted by V. The CP incurs a cost for maintaining the infrastructure (servers, bandwidth, software, etc) which increases with the infrastructure level I. This cost is denoted by $C(I)$. This cost typically increases with infrastructure level in a concave fashion because of economies of scale associated with content delivery. Finally, when the CP faces very high demand requests may be lost. The last term, *c.L(I,X)* denotes the cost of lost requests to the CP when its infrastructure level is I and it faces X requests in the given time period. c denotes the cost of a lost request[5] and $L(I)$ is the expected number of lost requests. That is, $L(I) = E[L(I,X)]$. The CP can choose low capacity (infrastructure) but will incur a high cost of lost requests or can choose a very high capacity level and incur fewer lost requests (but incur high infrastructure costs). The CP's decision problem is $\max_I \{U_{self}(I)\}$. Let us denote the optimal infrastructure level as I^* and associated expected surplus as $U_{self}(I^*)$.

The CP's surplus from delivering content through a CDN is given by - $U_{CDN}(X) = V(X) + \tau(N) \cdot X - C_O - P(X)$. Because a CDN delivers content intelligently to users based on their geographic location and network traffic, the CP derives a benefit from faster delivery of content. This may be in the form of improved customer retention rates or direct revenues. We assume that this benefit is linear in X, the number of requests served faster, with τ being the scaling factor. Further, larger the network of servers that the CDN maintains, the more likely that content is served faster to end users. Thus, τ increases with the CDN network size N. That is, a CDN with N servers provides a benefit of $\tau(N)$ to the CP from faster response for a single request. C_O denotes the transaction cost of outsourcing content delivery. For example, a CP with secure content may incur a cost in sharing the content with a third party (the CDN). This cost is assumed to vary across CPs. Finally, CDNs use usage based pricing model (price based on data delivered) and this is denoted as $P(X)$, the price for transmitting traffic of volume X in any given period[6]. Note that the CDN delivers entire websites from the edge with services such as Edgesuite and thus the CP does not need to maintain a large infrastructure for content delivery. Thus, the CP does not incur $C(I)$. Since the CDN has enough capacity to considerably minimize the lost requests, we approximate $L(I) = 0$. Once again, since a CP cannot precisely predict X in any period, it can compute the expected surplus $U_{CDN} = E[U_{CDN}(X)]$ and evaluate the CDN service accordingly. The CP will choose the CDN if $U_{CDN} \geq U_{self}(I^*)$. Based on these subscription decisions, one can evaluate the optimal price function *P(X)* for the CDN.

Let us start by analyzing how $U_{self}(I^*)$ may be determined and we will then proceed to determine the optimal price function.

4.1 Optimal Infrastructure Sizing

In the HTTP protocol, exchange of data between a server and client occurs after a TCP connection has been established. When a client attempts to establish a TCP connection, it begins by sending a SYN message to the server. The server acknowledges the SYN message by sending a SYN-ACK message to the client. In addition, the server creates a socket for the incoming connection and places it in the SYN-RCVD queue. Subsequently, the client responds with an ACK message. Upon receiving the ACK message, the server moves the corresponding socket to the accept queue. The connection between the client and the server is then open. Whenever a web server process is ready to respond to a connection request, it executes an accept() system call and receives a socket number from the accept queue in return. In other words,

[4] Note that X is different for each CP. We will not use the index *i* for simplicity but it will be implicit that X is indexed by content provider.

[5] This implicitly assumes that all requests are similar, which is a simplifying assumption. Given the granularity of the analysis, it suffices to assume an average cost per lost request.

[6] Some CDNs charge for usage by taking periodic samples of content delivered and then billing for the 95th percentile of usage. Our study uses the pure usage based pricing, wherein the CDN meters total content delivered (as opposed to sampling) and charges for the content delivered in that period.

requests are queued (the maximum size of the queue is determined by the kernel variable *somaxconn*) and wait for their turn to be processed. For further information on HTTP connection establishment, the reader is referred to Banga and Druschel (1997).

We model a web server as an *M/M/1/K* queuing system. That is, we assume that requests follow a poisson process with λ being the mean arrival rate (i.e., inter-arrival times for requests follows an exponential distribution with mean $1/\lambda$). We also assume that the service time of a request follows an exponential distribution. We model the service center as a single server system. Note that even a single web server entails several requests being simultaneously processed by different child processes. Furthermore, the architecture of most high end content delivery systems involves the use of server farms. Thus, a single server system is not an accurate representation of the system. However, we will show in supplemental tests that the general nature of the results remains the same with a multiple server model. Note also that our model of a service center, servicing requests at *I* requests/minute on average, folds the processor and bandwidth into a single service center. Thus we interpret infrastructure needed to handle *I* requests/minute as processors to handle *I* requests/minute and bandwidth to handle the corresponding amount of data outflow. Finally, we model a finite queue size of *K* requests, which is consistent with the observation that most commercial systems have similar settings for somaxconn and vendors recommend setting the queue size to somaxconn. Thus, we treat the queue length as an exogenous parameter that is determined by the vendor.

With the aforementioned abstraction of content delivery infrastructure, we now return to the CP's infrastructure sizing problem. The CP's net expected surplus from choosing infrastructure that can process *I* requests/minute on average is given by

$$U_{self} = V - C(I) - c \cdot L(I) \qquad (1)$$

We assume a quadratic cost function $C(I) = a \cdot I - b \cdot I^2$, which captures the concavity of infrastructure costs. A large value for *a* indicates high infrastructure costs and a large value for *b* indicates significant economies of scale. The cost function is defined in the region $I \le a/2b$, where costs are increasing in infrastructure level *I*. For an *M/M/1/K* queuing system, the expected number of lost requests is given by $L(I) = \dfrac{\lambda\left(1-\frac{\lambda}{I}\right)\left(\frac{\lambda}{I}\right)^K}{1-\left(\frac{\lambda}{I}\right)^{K+1}}$. A well known result in queuing theory indicates that *I* should be greater than λ, else the system "blows up"[7]. It can be verified that $\dfrac{\partial L(I)}{\partial I} < 0$ and thus the CP faces an optimization problem in trading off the benefits and costs of added infrastructure. Thus, the CP's decision problem is

$$\max_I \left\{ V - (a \cdot I - b \cdot I^2) - c \cdot \frac{\lambda\left(1-\frac{\lambda}{I}\right)\left(\frac{\lambda}{I}\right)^K}{1-\left(\frac{\lambda}{I}\right)^{K+1}} \right\} \qquad (2)$$

The associated first-order necessary condition is given by:

$$-a + 2b \cdot I - \frac{c \cdot \lambda^{K+1}}{I^{K+1} - \lambda^{K+1}} + \frac{c \cdot (K+1) \cdot \lambda^{K+1} (I-\lambda) I^K}{(I^{K+1} - \lambda^{K+1})^2} = 0 \qquad (3)$$

While it is difficult to evaluate the polynomial to compute closed form expressions for I^*, we can however deduce important properties of the solution by applying the conjugate pairs theorem from calculus (Currier 2000). The theorem states that for the maximization problem $\max_x F(x,a)$, the derivative $\dfrac{\partial x^*}{\partial a}$ and the cross partial F_{xa} have the same sign. The following results follow:

i) If the cost of infrastructure increases, I^* decreases.

$U_{Ia} = -1$. This implies that $\dfrac{\partial I^*}{\partial a} < 0.$ As expected, if the infrastructure costs (cost of processing and bandwidth) decrease, one expects content delivery infrastructure to be upgraded.

ii) If there are significant economies in scale in content delivery, I^* increases.

$U_{Ib} = 2I > 0$. Thus $\dfrac{\partial I^*}{\partial b} > 0.$ That is, if server vendors or bandwidth sellers provide higher volume discounts, infrastructure levels of content providers will increase.

iii) If a content providers cost of losing requests is high, I^* is correspondingly higher.

[7] The intuition is that if $I \le \lambda$, the mean arrival rate is greater than the mean service rate and the server keeps lagging further and further behind. Thus, our region of interest is $I > \lambda$. Further, since we have used a quadratic cost function, the region of interest is $I \in (\lambda, \frac{a}{2b}]$.

Proof:

$U_{Ic} = \frac{(K+1)\cdot\lambda^{K+1}(I-\lambda)I^K}{(I^{K+1}-\lambda^{K+1})^2} - \frac{\lambda^{K+1}}{I^{K+1}-\lambda^{K+1}}$. We know from the first order condition that $-\{a-2bI\} - \frac{c\lambda^{K+1}}{I^{K+1}-\lambda^{K+1}} + \frac{c(K+1)\lambda^{K+1}(I-\lambda)I^K}{(I^{K+1}-\lambda^{K+1})^2} = 0$. Since $-\{a-2b\cdot I\} < 0$ (rate it which infrastructure costs increase with infrastructure level *I*), it follows that $-\frac{c\cdot\lambda^{K+1}}{I^{K+1}-\lambda^{K+1}} + \frac{c\cdot(K+1)\cdot\lambda^{K+1}(I-\lambda)I^K}{(I^{K+1}-\lambda^{K+1})^2} = cU_{Ic} > 0\cdot$

Since $c > 0$, it follows immediately that $U_{Ic} > 0$. From the conjugate pairs theorem, $\frac{\partial I^*}{\partial c} > 0.$ In other words, if the cost (c) of losing a request increases, the optimal infrastructure level also increases.

iv) If the arrival rate of requests λ increases, I^* increases.

Proof: The above statement follows from conjugate pairs theorem if $U_{I\lambda} > 0$ is true. Upon computing the cross partial with respect to I and λ, it can be shown that $U_{I\lambda} > 0$ iff

$K(I-\lambda)(I^{K+1}+\lambda^{K+1}) - 2I\lambda(I^K-\lambda^K) > 0$. This can be restated as: $U_{I\lambda} > 0$ iff

$Kp^{K+2} - (K+2)p^{K+1} + (K+2)p - K > 0$ (A)

where $p = I/\lambda$. For $I >> \lambda$, it follows that $p >> 1$. Thus

$(K+2)p > K$ (B)

Also, for large K (queue size), we know the following is true: $p > 1 + \frac{2}{K}$. This may be restated:

$Kp^{K+2} > (K+2)p^{K+1}$ (C)

Adding (B) and (C), we get: $Kp^{K+2} + (K+2)p > (K+2)p^{K+1} + K$.

Combining this result with (A), it follows that $U_{I\lambda} > 0$. That is, $\frac{\partial I^*}{\partial \lambda} > 0.$ QED.

All the above results conform to intuition. It is interesting to see how exactly the optimal infrastructure level changes with request arrival rate λ. To determine the nature of this relationship, we use numerical techniques due to the limitation of the analytical model in deriving a closed form solution.

For the infrastructure cost function, $C(I) = a\cdot I - b\cdot I^2$, we assume that *a=3.46* and *b=0.000043*. This implies that the cost of serving 233 requests/min is $804 per month. If we assume that the average size of the response to a request is 50 Kbytes, this implies that the cost of serving data at 1.55 Mbps is $804 p.m. This is reasonable given the cost of a T1 connection (approx. $400 p.m) and maintaining a workstation. The cost of serving 6975 requests/min is $22042. This corresponds to a T3 connection and maintaining a server. The cost of serving 23,255 requests/min (equivalent to an OC3 connection) is $57208. These costs are also comparable to managed hosting costs today. We assume that the cost of a lost request, *c*, is $10. This is based on an assumption that 10% of visitors purchase products/services, the average purchase is $100 and a customer leaves a website if a request does not go through. Finally, we assume that the queue size, *K*, (for requests waiting to be processed) is 8 requests. Note that we have set the cost of lost requests high and queue size low in order to eliminate boundary solutions where $I^* = \lambda$. This is because we are interested in the nature of the relationship for interior solutions. Figure 1 shows the optimal infrastructure level (in requests/min) for different arrival rates ranging from 5000-20,000 requests/min. The relationship is approximately linear.

To test for robustness, we repeated the numerical analysis for a wide variety of settings and found that the relationship is near linear in all cases. Figure 2 shows the relationship for the case where $\{a = 3.46; b = 0.000043; K = 4; c = 20\}$. The relationship is again approximately linear. Note that the boundary solution cases of $I^* = \lambda$ are also linear.

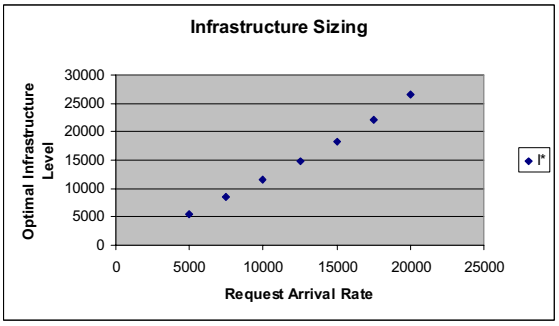

Figure 1. Optimal Infrastructure Level versus Arrival Rate (Case 1)

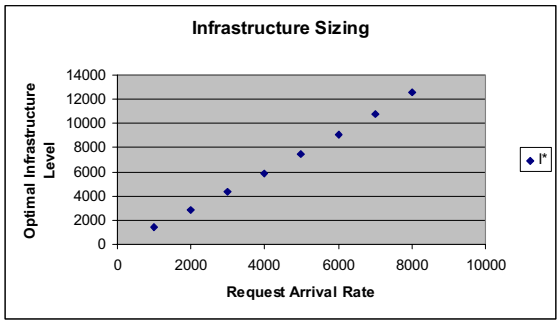

Figure 2. Optimal Infrastructure Level versus Arrival Rate (Case 2)

Multiple Servers

The analysis thus far has been based on an abstraction of request servicing as a single server system. In this subsection, we relax that assumption and numerically evaluate the characteristics of a multiple server system. We analyze a case with three servers instead of one. The queue size is assumed to be 5, cost of a lost request is $10 and the remaining settings are as before,i.e., $\{a = 3.46; b = 0.000043; K = 5; c = 10\}$. The optimal infrastructure level (in requests/min) for different arrival rates is shown in figure 3 below. The relationship continues to be linear. As is intuitive, the optimal infrastructure level for each server (I^*) can now be lesser than the mean arrival rate, λ, as three servers are sharing the load. We shall now proceed to determine the optimal price that the CDN must charge.

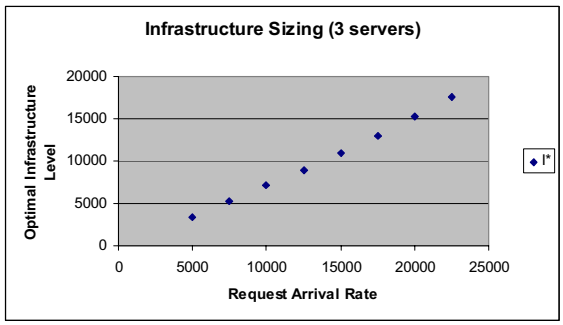

Figure 3. Optimal Infrastructure Level Vs. Arrival Rate (multiple servers)

4.2 CDN Pricing Problem

As stated earlier, the CP's surplus from choosing a CDN is given by

$$U_{CDN}(X) = V(X) + \tau(N) \cdot X - C_O - P(X) \quad (4)$$

The CP does not know exactly how many requests (X) will be made for its content in any period. The CP's expected surplus from choosing the CDN is

$$U_{CDN} = V + \tau(N) \cdot \lambda - C_O - E[P(X)] \quad (5)$$

Given any price function *P(X)*, the CP can compute its expected surplus. The CP chooses the CDN if $U_{CDN} \geq U_{self}(I^*)$. Substituting equations 1 and 5, a CP with arrival rate λ subscribes to the CDN if

$$C_O \leq \tau(N) \cdot \lambda + C(I^*) + c \cdot L(I^*) - E[P(X)] \quad (6)$$

Note that the outsourcing cost C_o varies across CPs. *H()* denotes the cdf of C_o and *h()* denotes the pdf. Thus, the probability that a CP with mean arrival rate λ subscribes to a CDN is given by $H\left(\tau(N) \cdot \lambda + C(I^*) + c \cdot L(I^*) - E[P(X)]\right)$. If $g(\lambda)$ denotes the number of CPs with mean arrival rate λ, then the expected number of these CPs subscribing to the CDN is given by $g(\lambda)H\left(\tau(N) \cdot \lambda + C(I^*) + c \cdot L(I^*) - E[P(X)]\right)$. The CDN's expected profit, with the standard assumption of zero marginal cost, is given by

$$\pi = \int_{\lambda} g(\lambda) \cdot H(\tau(N) \cdot \lambda + C(I^*) + c \cdot L(I^*) \cdots \\ \cdots - E[P(X)]) \left(\int_X P(X) \cdot \left(\frac{e^{-\lambda} \lambda^X}{X!} \right) dX \right) d\lambda \quad (7)$$

The CDN chooses the price function *P(X)* in order to maximize π. Note that the CP does not know the value of X that will be realized in any period and thus makes the subscription decision based on *E[P(X)]* (as can be verified from equation 6). Similarly, the CDN computes its expected profit by evaluating *E[P(X)]* for each subscribing CP. Thus, the CDN can achieve the same subscription levels and expected profits by charging a fixed amount *E[P(X)]* per period to each CP. This is also illustrated in figure 4. Consider the optimal price function, denoted by $P^*(X)$. Consider a CP with mean arrival rate λ_1. In each period, the CP receives a random X number of requests and pays *P(X)* in that period. Over a long period of time, the CP expects to receive a mean of λ_1 requests per period and pays *E[P(X)]*. In fact, if the CDN offers an alternative pricing scheme, wherein it charged the CP a fixed amount *E[P(X)]* per period, the CP would still make the same subscription decision. Similarly, the CDN could charge all CPs with arrival rate λ_2 the corresponding expected price of *E[P(X)]* per period as shown in the figure (*E[P(X)]* is different for CPs with mean λ_1 and λ_2). Thus, for any optimal price function $P^*(X)$, there exists a corresponding "mean-usage-based" price function P_λ, obtained by following the trajectory of *E[P(X)]* for different values of λ, that achieves the same results. We can use this observation to simplify the problem to that of determining the optimal P_λ and then determining a corresponding P(X).

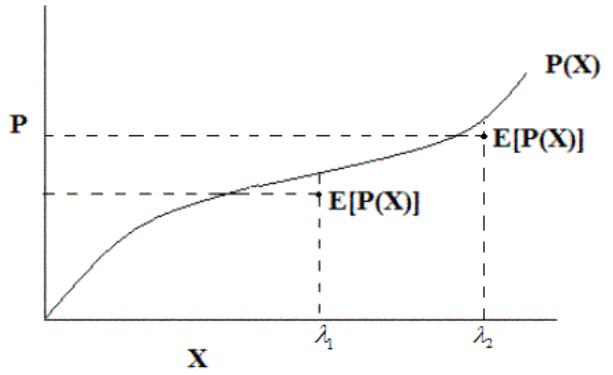

Figure 4. Pure usage based price and mean price

Consider CPs with mean arrival rate λ_1. The CDN charges all such CPs a fixed price $P_{\lambda 1}$. The CDN's expected profit from these CPs is given by $\pi =$

$$g(\lambda 1)H\left(\tau(N)\cdot\lambda + C(I^*) + c\cdot L(I^*) - P_{\lambda 1}\right)\left(P_{\lambda 1}\right) \quad (8)$$

The optimal price $P_{\lambda 1}$ obtained directly by applying the necessary first order condition is given by the solution to the following equality:

$$P_{\lambda 1} = \frac{H\left(\tau(N)\cdot\lambda_1 + C(I^*) + c\cdot L(I^*) - P_{\lambda 1}\right)}{h\left(\tau(N)\cdot\lambda + C(I^*) + c\cdot L(I^*) - P_{\lambda 1}\right)}.$$

The optimal "mean-usage-based" price function, P_λ, is thus given by

$$P_\lambda = \frac{H\left(\tau(N)\cdot\lambda + C(I^*) + c\cdot L(I^*) - P_\lambda\right)}{h\left(\tau(N)\cdot\lambda + C(I^*) + c\cdot L(I^*) - P_\lambda\right)} \quad (9)$$

A special case with uniform distributions: To illustrate a few properties of the optimal price function, we make the following two assumptions – (1) the outsourcing cost, C_o, is uniformly distributed in [0,1]. That is, *H(x) =x* and *h(x) = 1*. (2) The optimal infrastructure level I^* for a CP with mean arrival rate λ is given by $I^* = i_o\lambda$, where i_o is a constant. We shall soon relax assumption 1 to test the impact on our results. Assumption 2 can be justified by the numerical results in section 4.1 (see figures 1 and 2). With these assumptions, the optimal price function obtained by simplifying equation (9) is as follows:

$$P_\lambda = \frac{1}{2}\left[\tau(N) + ai_o + \frac{c(i_o - 1)}{i_o^{M+1} - 1}\right]\lambda - \frac{b\cdot i_o^2}{2}\lambda^2 \quad (10)$$

A usage-based price function, $P^*(X)$ for which the *E[P(X)]* trajectory is given by equation 10 is:

$$P(X) = \frac{1}{2}\left[\tau(N) + ai_o + bi_o^{\ 2} + \frac{c(i_o - 1)}{i_o^{M+1} - 1}\right]X - \frac{b\cdot i_o^2}{2}X^2 \quad (11)$$

To verify that equation 11 represents the optimal usage-based price function, let's assume that there is a different price function, $P^A(X)$ that performs better than *P(X)*, i.e., yields higher expected profit. In that case, the corresponding "mean-usage-based" price function represented by $E[P^A(X)]$ should also provide higher expected profit than *E[P(X)]*. This cannot be true since equation 10 represents the optimal "mean-usage-based" price. Thus, our assumption is wrong implying that equation 11 represents the optimal usage-based price function.

The following observations can be made regarding the optimal pricing policy:

a) Volume discounts: It can be verified that $\frac{\partial P(X)}{\partial X} > 0$ and $\frac{\partial^2 P(X)}{\partial X^2} < 0$ for the relevant range of *X*. Thus, the optimal pricing policy entails volume discounts to CPs. This is, in fact, consistent with Akamai's pricing statement:

"...Customers commit to pay for a minimum usage level over a fixed contract term and pay additional fees when usage exceeds this commitment. Monthly prices currently begin at $1,995 per megabit per second, with discounts available for volume usage."

Equation (11) indicates that the volume discounts essentially follow from the economies of scale in content delivery costs (*b>0*). Higher the volume discounts that bandwidth sellers offer, greater is the discount that CDNs will have to offer.

b) Larger CDN networks are able to charge higher prices in equilibrium. It can be verified $\frac{\partial P}{\partial N} > 0$. This is an intuitive result.

c) $\frac{\partial^2 P}{\partial N \partial X} > 0$. That is, a larger CDN can extract a higher increase in price than a smaller CDN, for the same increase in volume of traffic. That is, given that the amount of traffic handled by CPs is on the rise, a larger CDN can leverage this trend better by having a higher increase in price than a smaller CDN.

d) A CP with mean arrival rate λ subscribes to the CDN if $C_o \le \frac{1}{2}\left[\tau(N) + Ai_o + \frac{c(i_o - 1)}{i_o^{M+1} - 1}\right]\lambda - \frac{B\cdot i_o^2}{2}\lambda^2$.

This is obtained by substituting the optimal price function

into the subscription condition in equation (6). This subscription decision is shown in the figure below. As can be seen, for any given C_o, CPs with high λ will subscribe to the CDN. Similarly, for any given λ, CPs with low C_o will subscribe. In other words, CPs expected to subscribe to a CDN are those with high volume of traffic and low content delivery outsourcing cost (for example, content with minimal security requirements).

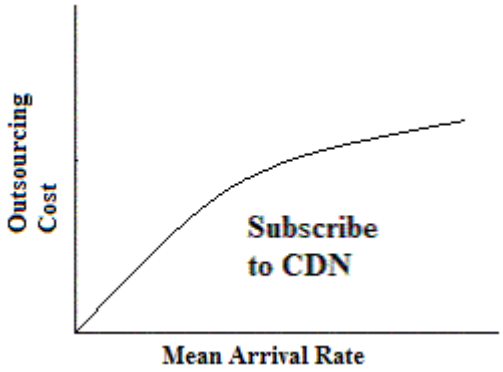

Figure 5. CP subscription decision

e) To test for the impact of distributional assumptions, we solved the same model but introduced a skew in the distribution of outsourcing cost across CPs by setting the distribution as follows: $H(C_o) = C_o^{\,2}; f(C_o) = 2C_o$. Relative to the uniform distribution, this distribution assumes that there are relatively more CPs with high outsourcing cost i.e., content requiring high security (see figure 6). The new solution is given by

$$P(X) = \frac{1}{3}\left[\tau(N) + Ai_o + \frac{c(i_o - 1)}{i_o^{M+1} - 1}\right]X - \frac{B \cdot i_o^2}{3}X^2.$$

This price is lower than the optimal price in the case of the uniform distribution. Thus, the price charged decreases as the number of CPs with high outsourcing costs increases. The converse is true when there are relatively fewer CPs with high outsourcing costs. In the light of increasing security concerns in content delivery, CDN prices will have to decline unless improvements in technology can address the security costs. Other forms of outsourcing costs may include cost of interfacing with a third party, gathering business intelligence or modifying content in order to enable delivery from a third party. As an example, switching to Akamai would require a CP to make its content ESI (Edge Side Includes – a technology developed by Akamai to enable edge delivery) compatible. Thus, the technology choice for redirection (URL rewriting or DNS redirection) is a critical one because URL rewriting may impose higher outsourcing costs on CPs. The CDN thus needs to consider the imposed outsourcing costs in addition to efficiency-based technical evaluation of redirection schemes.

f) As bandwidth, memory and processor costs decline, the price that the CDN can charge will also decrease. Thus, the declining infrastructure costs will result in lower revenues for CDNs. Note that these declining costs also reduce the infrastructure costs for the CDN. Numerical results suggest that the net impact will be a decline in CDN profits.

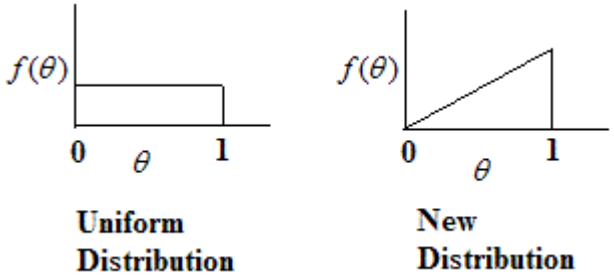

Figure 6: Negative skew in distribution of outsourcing cost

4.3 Relaxing the Assumptions

The model presented in sections 4.1 and 4.2 assume that requests for content at a web server follows a Poisson arrival process. Recent traffic engineering studies (for e.g., Barford and Crovella 1998) have demonstrated that a self similar process is more accurate in describing the characteristics of requests.

Definition: For any timeseries, X(t), the m-aggregated series X(m) is obtained by summing the original series over nonoverlapping blocks of size m. Then X is self similar if for all m, X(m) has the same distribution as X rescaled by m^H, where H is a constant known as the Hurst parameter of the distribution. X has the same autocorrelation function $r(k) = E[(X_t - \mu)(X_{t+k} - \mu)]/\sigma^2$ as X(m).

For the purposes of our model, there are two important properties of self similar processes to note. First, the autocorrelation function of a self similar process decays hyperbolically as opposed to an exponential decay for Poisson processes, i.e., $r(k) \sim k^{-\beta}$. This property is also known as long-range dependence. Second, self similar traffic demonstrates significantly higher burstiness than Poisson traffic.

Two other important parameters in our model are the distribution of mean arrival rates across CPs (functional form of $g(\lambda)$) and the distribution of outsourcing costs across CPs ($h(C_o)$). While it is difficult to estimate the latter (can be done by conducting

surveys), the functional form of $g(\lambda)$ can be obtained by analyzing appropriate traces. In this paper, we use the IRCache trace, obtained from NLANR's IRCache project (NLANR 2002). NLANR maintains a network of ten medium/large caches at various locations in the US. Caches at other organizations as well as individual browsers of end users are reconfigured to redirect requests to one of the NLANR caches. Traces in this analysis were recorded at the cache in Boulder, CO on September 19 and September 23, 2002. Since IRCache traces have been extensively studied and benchmarked in the web caching literature, we treat the sample of requests in our trace as representative of the generic Internet traffic.

Figure 7 plots the histogram of the mean arrival rate for different websites on a log-log scale. Tests suggest that the distribution of λ across CPs is well approximated by a Pareto distribution. Thus, we can model $g(\lambda) = c1/\lambda^{c2}$, where *c1* and *c2* are constants.

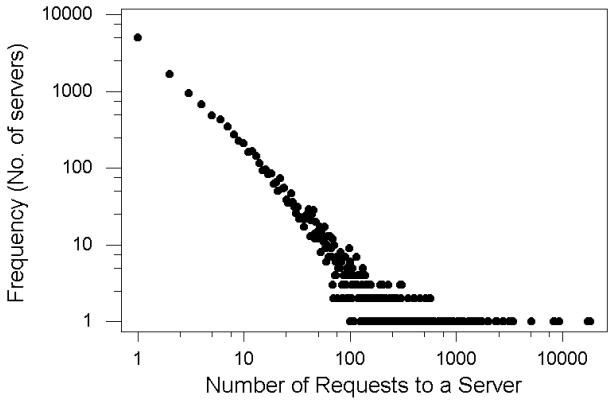

Figure 7. Histogram of mean number of requests to a server

The optimal infrastructure level with pareto distribution of mean arrival rates across servers and self similar traffic at individual servers is shown in figure 8 (the plot for Poisson traffic is also shown for reference). The remaining settings for this numerical example are the same as used in figure 1. As expected, the optimal infrastructure level is higher for self similar traffic than Poisson because of higher burstiness. Furthermore, the infrastructure level increases with the mean arrival rate at a faster rate than for Poisson arrivals. That is, the error from using a Poisson process increases with intensity of traffic. This is because the burstiness of Poisson traffic decreases with intensity and thus it underestimates the true traffic burstiness even more with higher intensity traffic. The CDN is able to charge higher prices than in the case of Poisson traffic. Further, the extent of the volume discount is lower (but it continues to exist) than in the case of Poisson distribution because the Poisson model underestimates the burstiness of high volume traffic. Increased burstiness will likely increase the fixed infrastructure costs of the CDN as well (not considered in our model).

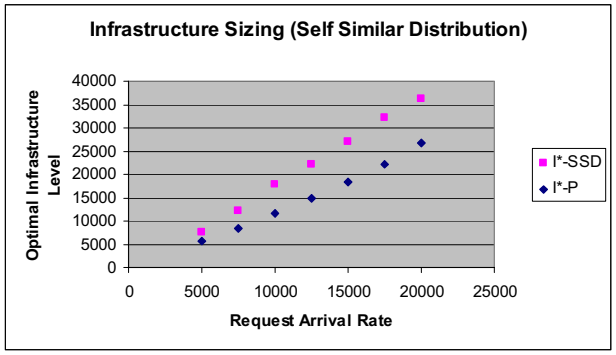

Figure 8. Infrastructure Sizing (Self Similar Traffic and Poisson Traffic)

5 Conclusions

Content Delivery Networks are an important element of the Internet supply chain. These services bring content closer to consumers and allow content providers to hedge against highly variable traffic. Pricing of these services is a challenging problem faced by managers in CDN firms. Deployment of new services such as Edgesuite that enable delivery of entire websites from the edge are accompanied with open questions regarding pricing and service adoption.

We develop an analytic model to analyze optimal pricing policies for CDN operators. Our model shows that, consistent with industry practices, CDN pricing functions should provide volume discounts to content providers. They also show that the most likely subscribers to CDN services are those content providers with high volume of traffic and low security requirements. Significantly, our model also shows that larger CDN networks can charge higher prices in equilibrium, which should strengthen any technology-based economies of scale and make it more difficult for entrants to complete against incumbent firms. Given that Akamai enjoys 80% of the market share, it will be difficult for entrants to challenge the dominance of Akamai. We also find that the price charged by CDNs will decline if security concerns associated with content delivery on the Internet increases. Various studies and surveys show that security is the prime content delivery related concern of CPs supporting the notion that CDNs will need to invest in developing solutions that allay such concerns. Finally, we find that declining bandwidth costs will negatively impact the revenues and likely the CDN profit as well.

6 References

[1] P. Barford and M. E. Crovella. Generating representative web workloads for network and server performance evaluation. Proceedings of ACM SIGMETRICS, July 1998.

[2] G. Banga and P. Druschel. Measuring the capacity of a Web server under realistic loads. *World Wide Web Journal (Special Issue on World Wide Web Characterization and Performance Evaluation)*, 2(1), May 1999.

[3] Y. Chen, R. Katz, and J. Kubiatowicz. Dynamic Replica Placement for Scalable Content Delivery. In Proceedings of the First International Workshop on Peer-to-Peer Systems (IPTPS) 2002.

[4] R. Cocchi, S. Shenker, D. Estrin, and L. Zhang. Pricing in computer networks: Motivation, formulation and example. *IEEE/ACM Transactions on Networking*, vol. 1, December 1993.

[5] K. Currier. *Comparative Statics Analysis in Economics*. World Scientific Publishing Company. August 2000.

[6] A. Datta, Kaushik Datta, Helen Thomas, Debra VanderMeer. WORLD WIDE WAIT: A Study of Internet Scalability and Cache-Based Approaches to Alleviate it. Georgia Institute of Technology Working Paper.

[7] J. Dilley, Bruce Maggs, Jay Parikh, Harald Prokop, Ramesh Sitaraman, and Bill Weihl. Globally Distributed Content Delivery. *IEEE Internet Computing*. Sep-Oct 2002.

[8] S. Gadde, Jeff Chase, and Michael Rabinovich. Web caching and content distribution: A view from the interior. In Proceedings of the Fifth International Web Caching and Content Delivery Workshop, Lisbon, Portugal, May 2000.

[9] K. Hosanagar, R. Krishnan, J. Chuang, and V. Choudhary. Pricing Vertically Differentiated Web Caching Services.Proceedings of the International Conference on Information Systems (ICIS), Barcelona, December 2002.

[10] K. Johnson, John Carr, Mark Day, and M. Frans Kaashoek. The measured performance of content distribution networks. In Proceedings of the Fifth International Web Caching and Content Delivery Workshop, Lisbon, Portugal, May 2000.

[11] B. Krishnamurthy, C. Wills, and Y. Zhang. On the use and performance of content distribution networks. ACM SIGCOMM Internet Measurement Workshop, 2001.

[12] B. Maggs, Vice President, Akamai. Personal Communication. 2002.

[13] Mendelson, H., and S. Whang. Optimal Incentive-Compatible Priority Pricing for the M/M/1 Queue. *Operations Research*, 38, 870-83, 1990.

[14] National Laboratory for Applied Network Research (NLANR). Ircache project. www.ircache.net/, 2002.

[15] M. Nottingham. On defining a role for demand-driven surrogate origin servers. In Proceedings of the Fifth International Web Caching and Content Delivery Workshop, Lisbon, Portugal, May 2000.