Pricing and Resource Provisioning for Delivering E-Content On-Demand with Multiple Levels-of-Service

Srinivasan Jagannathan and Kevin C. Almeroth

Department of Computer Science, University of California, Santa Barbara, CA 93106-5110
{jsrini,almeroth}@cs.ucsb.edu

Abstract. Businesses selling multimedia rich software or *e-content* are growing in the Internet. The e-content can be downloaded by the customer or alternately, streamed by the content provider, immediately after on-line transactions. Since Internet connection speeds are variable, ranging from dial-up access speeds to broadband speeds, the content providers may provide different *levels-of-service (LoS)* for the same content. If a provider offers service at different LoS, for example at 56 kbps, and 128 kbps, how should the price of the service be set such that the provider makes the most money? In addition, how should the resources be provisioned among the different service levels? In this paper, we address such pricing and resource provisioning issues for delivering e-content at multiple service levels.

1 Introduction

Available bandwidth and usage have increased in the Internet. Use of the Internet to purchase goods and services is also increasing. At the same time, the multimedia capabilities of computers are improving while remaining affordable. Together, these trends have spawned services offering video-on-demand, downloadable CDs etc. While such services are growing, all users do not have the same Internet connection speeds. Some users connect at dial-up speeds while others at broadband speeds. To accommodate this heterogeneity in connection speeds, the content provider may serve the same content at different quality levels. For instance, many web sites offer the same streaming content at two quality levels—56 kbps and 128kbps. Since requests for different levels-of-service (LoS) consume different amount of server resources and are presumably quoted a different price, one must examine how to provision resources for each LoS. In this paper, we seek to answer the following questions: 1) how should a content-provider price content at each LoS, and 2) how should a content provider allocate resources for each LoS.

In this work, we differentiate between connectivity and content pricing. We focus on the latter, i.e., our work does not deal with pricing the customers' or the content provider's Internet connection. Instead, it deals with how much the customers pay for content. We assume that both the customer and the content

provider have suitable Internet connectivity to participate in the transaction. Our work builds on our earlier research for pricing on-demand delivery of content when there is single LoS [1–4]. In our earlier work [5], we compared a number of simple pricing schemes using simulations. These pricing schemes could be classified as being *static* or *dynamic*. In a static pricing scheme, the price of the content does not change frequently. In a dynamic pricing scheme, the price may vary on much smaller time scales based on factors like current server load, request arrival rate etc. In our simulations, we observed that static pricing schemes do not perform well when insufficient information is known about the customer population. Based on the simulations, we believe that there exist fixed prices which generate very high revenues but, finding these prices is non-trivial. Furthermore, the fixed prices that generate highest revenue can differ based on the customer population, request arrival patterns, server load etc. We formulated a dynamic pricing scheme called HYBRID which not only generated consistently high revenues across a range of simulation scenarios and customer populations, but also reduced the number of requests rejected due to lack of server resources. In this paper, we primarily focus on extending the HYBRID pricing scheme to systems with multiple levels-of-service (multi-LoS systems). We validate our work through simulations.

There are two challenges in designing pricing schemes for multi-LoS systems. First, it is difficult to quantify the "capacity" of the system. For instance, consider a system where resources are quantified in terms of channels. Consider a system with 100 channels and two LoS. Suppose that for a lower LoS one channel is allocated, and for a higher LoS two channels are allocated. Then the system can accomodate 100 low LoS requests or 50 high LoS requests. The actual number of requests that the system serves will vary with the relative fraction of high vs low LoS requests. Moreover, when there are more requests than the system can serve, it is difficult to decide which requests to satisfy. For instance, if it is known that customers are willing to pay at least \$5 for the lower LoS and \$7 for the higher LoS, accepting low LoS requests will increase the revenue when resources are constrained. However, since how much customers are willing to pay is not known, deciding which requests to serve is difficult.

The other challenge in pricing multi-LoS systems is in understanding customer behavior. For customers with high bandwidth connections, the choice of LoS depends depend not only on the LoS actually desired but also on how other LoS for that content are priced. For instance, suppose that a customer with a high bandwidth connection is willing to pay \$9 for a low LoS. If the desired LoS is priced at \$6 and the higher LoS at \$8, then the customer may choose the higher LoS. Though this increases the revenue by \$2, it may prevent another low LoS request from being satisfied. The system loses \$4 in this case. In this paper, we make a simplifying assumption that the customers choice of LoS is independent of the price for other LoS. This is a reasonable assumption because in the Internet today, content is typically served at LoS where there is a perceptible difference in quality between the LoS. Customers with high bandwidth connections may typically not purchase content at low LoS. We shall address the general problem where choice of LoS is correlated with price in future work.

We briefly survey related work in the following. Basu and Little[6], have formulated models for VoD and pricing issues related to them. Mackie-Mason et al.[7] investigate adaptation to changes in consumer variables for an information

goods market. Sairamesh and Kephart [8] discuss competition and price wars in information goods markets. Their analysis assumes that each competitor sells at a different LoS. All the above do not consider distribution constraints of the content provider. Wolf et al. [9] study how to maximize profits when broadcasting digital goods. When resources are constrained, they schedule the delivery at a later time, and pay a penalty for late delivery by charging a lower price. Chan and Tobagi [10] design scheduling schemes for batched delivery of video-on-demand, when the fixed price for the content is known. Their work does not consider multiple levels-of-service. To the best of our knowledge, though there has been considerable work on connectivity pricing, there has been very little work on pricing on-demand delivery of e-content when there are constraints on the distribution resources of content providers.

The rest of the paper is organized as follows. Section 2 describes a formulation for revenue earned in multi-LoS systems. Section 3 describes our HYBRID pricing scheme and two other dynamic pricing schemes adapted from the work by Sairamesh and Kephart [8]. Section 4 discusses the simulation framework and the experiments we perform. Results are presented in Section 5. We conclude the paper in Section 6.

2 Revenue Model and Resource Provisioning

We consider a system where requests are satisfied if resources are available and the customer agrees to pay the quoted price. It is assumed that all the server resources can be quantified and mapped to a real number. One approach to doing this is to consider the bottleneck resource at the server as the indicator of system resources. For example, if bandwidth[1] is the bottleneck, then the total available bandwidth is modelled as the system capacity. For the purposes of this paper, we shall assume that available connection bandwidth of the content provider is the measure of system resources. When a request is served, some of the connection bandwidth is allocated to that request[2]. Requests are processed on a First-Come-First-Served basis. If there is insufficient bandwidth available when a request arrives, then the request is rejected. In our model, we assume that once the content provider makes the initial infrastructural investment, there are either negligible or fixed costs in maintaining the resources (caches, servers, bandwidth etc.), i.e., there are no additional costs based on number of requests served. This is a reasonable assumption because servers incur fixed costs and bandwidth can be bought at a flat monthly rate. If maintenance costs are negligible or fixed, profit maximization is equivalent to revenue maximization. We also assume that the market is monopolistic, i.e., there is no other entity selling the same content. This is a realistic assumption in many scenarios where the content owner personally sells the content or has licensed it to a single distributor.

Table 1 presents the symbols we have used in our analysis. Since we assume that a customer's choice of LoS is independent of price, we can treat the same content at different LoS as different products. Consider an arbitrary customer who wants to purchase content $p_{i,j}$. We denote his/her decision to purchase the service by the random variable $\Upsilon_{i,j}$ which can take two values, 1 for accept and 0 for reject. Let $E[\Upsilon_{i,j} \mid \psi_{i,j}]$ denote the expectation of the decision to purchase

[1] Other bottlenecks include memory, and latency.

[2] This does not imply that network resources are reserved.

Notation	Description
m	Number of products
L	Number of levels of service
$\mathcal{B}$	Total system resources
b_j	Resources provisioned for j^{th} LoS
l_j	Resources for serving a request at j^{th} LoS
$p_{i,j}$	i^{th} product at j^{th} LoS
$\Upsilon_{i,j}$	Decision to purchase $p_{i,j}$ (0 or 1)
$\psi_{i,j}$	Price of $p_{i,j}$
$\lambda_{i,j}$	Request arrival rate for $p_{i,j}$
$\mathcal{R}$	Total revenue per unit time
d	Mean service time
ρ	System Utilization

Table 1. Symbols Used

content $p_{i,j}$ when the price is $\psi_{i,j}$. The expectation of revenue per unit time is given by:

$$\mathcal{R} = \sum_{i=1}^{m} \sum_{j=1}^{L} \lambda_{i,j} \psi_{i,j} E[\Upsilon_{i,j} \mid \psi_{i,j}] \tag{1}$$

Notice that the revenue function described above does not consider resource constraints. To model resource constraints, we use the notion of system utilization. System utilization, ρ, is the relative fraction of time for which the channels are busy servicing requests. It is defined as the ratio of the number of requests entering the system per unit time to the number of serviced requests exiting the system per unit time. In a stable system, this ratio must be less than or equal to 1. Notice that, the number of requests that can be serviced per unit time depends on the system resources. If there are more requests than the system can serve, the predicted system utilization exceeds 1. In the revenue maximization problem in Equation 1, we impose an additional constraint that the predicted system utilization should be less than or equal to 1.

System utilization is easily defined when there is a single LoS. If l is the resources consumed by a request at this LoS, the system utilization can be computed as $\frac{dl}{\mathcal{B}} \sum_{i=1}^{m} \lambda_i E[\Upsilon_i \mid \psi_i]$. However, with multiple LoS, and requests at each LoS consuming different amount of resources, it is not possible to quantify the number of serviced requests exiting the system. We therefore take a different approach. Suppose that the system resources are partitioned into $\langle b_1, b_2, ..., b_L \rangle$, where b_j is the resource provisioned for level j. Then, we can impose the system utilization constraint independently for each LoS. We solve an independent constrained maximization problem for each LoS. The total revenue earned critically depends on how the resources are partitioned for each level. Notice that resources consumed by requests for level j will be less than or equal to $\sum_{i=1}^{m} l_j \lambda_{i,j}$. Based on this, we provision resources as follows: $b_j = \frac{\sum_{i=1}^{m} l_j \lambda_{i,j}}{\sum_{j=1}^{L} \sum_{i=1}^{m} l_j \lambda_{i,j}}$ The revenue maximization problem is then given by:

– Maximize: $\sum_{j=1}^{L} \mathcal{R}_j$ where $\mathcal{R}_j = \sum_{i=1}^{m} \lambda_{i,j} \psi_{i,j} E[\Upsilon_{i,j} \mid \psi_{i,j}]$
– Subject to:
  • $\psi_{i,j} \geq 0$, $1 \leq i \leq m$, $1 \leq j \leq L$
  • $\rho_j \leq 1$, $1 \leq j \leq L$, where $\rho_j = \frac{dl_j}{b_j} \sum_{i=1}^{m} \lambda_{i,j} E[\Upsilon_{i,j} \mid \psi_{i,j}]$

As can be observed, the revenue model relies on knowledge of the request arrival rate and the expectation of the decision to purchase, given the price. The request arrival rate can be monitored. However the expectation of the decision to purchase is not known. In the next section we outline the HYBRID scheme which estimates the expectation of the decision to purchase.

3 Dynamic Pricing Algorithms

In this section, we briefly describe the HYBRID pricing scheme. The HYBRID algorithm is based on the premise that customers are rational human beings. We are interested in the fraction of requests that will result in successful transactions. For a rational customer population, it can be argued that this fraction is a non-increasing function of the quoted price. For a price x, let $f(x)$ denote the fraction of customers who will accept the price. Let x_{low} be a price below which $f(x)$ is exceptionally high, say more than t_h and let x_{high} be a price above which $f(x)$ is exceptionally low, say below t_l. Then $f(x)$ can be approximated in the domain $[x_{low},\ x_{high}]$ using some non-increasing function. We propose a family of decreasing functions which depend on a parameter δ described as follows.

$$f(x) = \begin{cases} t_h & , \quad 0 \leq x < x_{low} \\ (t_h - t_l)\left[1 - \left(\frac{x - x_{low}}{x_{high} - x_{low}}\right)^{\delta}\right] + t_l & , \quad x_{low} \leq x \leq x_{high} \\ t_l & , \quad x > x_{high} \end{cases} \tag{2}$$

Figure 1 illustrates the family of non-increasing functions. By experimenting with different prices to observe the fraction of customers who accept the price, and using statistical methods like least squared errors, one can estimate the parameter δ, and the threshold prices x_{low} and x_{high}. Notice that $f(x)$ is also the expectation of the decision to purchase, given price x. Once all the parameters are known, the content provider can predict the customer behavior and thereby choose a price[3] using the optimization problem described in the previous section. In HYBRID, the customer reaction is continuously monitored, and the price is varied at regular intervals[4]. The details of this algorithm are presented in our earlier work [5].

On performing simulations with the scheme described above, we observed that while the revenue earned was high, the number of requests rejected due to lack of resources was also high. This was mainly because when the algorithm experimented with low prices, more customers accepted the service than could be accommodated by the server. We therefore modified the algorithm as follows. Whenever the server load increased beyond a certain threshold, an exponentially

[3] The price so obtained may not be the global optimum.
[4] Temporal price variations are an inherent feature in many commodity markets.

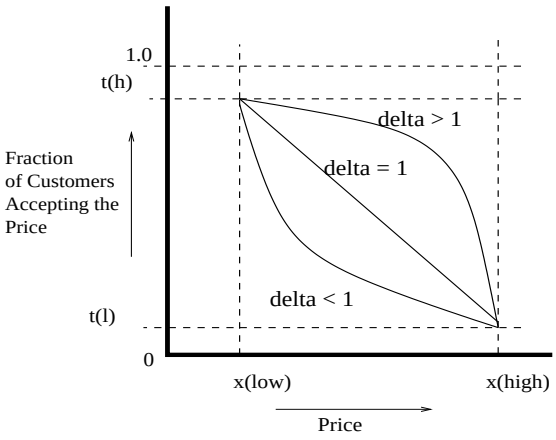

Fig. 1. Customer Model

increasing price was quoted. Suppose that x is the fraction of available resources that have been allocated to satisfy requests. Let L and H be the lowest price and highest price that the content provider decides to quote to customers. Then, if x is greater than a threshold, the price quoted to a customer, irrespective of the content requested, is given by: $(L-1) + (H-L+1)^x$. This modified algorithm, generated consistently high revenues while at the same time minimizing the number of requests rejected due to lack of resources.

Other Dynamic Pricing Algorithms

We present two dynamic pricing algorithms adapted from the work by Sairamesh and Kephart [8]. These algorithms were observed to converge to the game theoretic optimal price in a competitive market in the simulations performed by Sairamesh and Kephart[5]. We chose these algorithms for the purposes of evaluation and comparison with our algorithm.

Let L and H be the lowest and highest prices respectively that the content provider decides to quote. In the Trial-and-Error-Pricing (TEP) algorithm, an initial price is chosen at random in the range $[L, H]$. At regular intervals, with a small probability (called small jump probability) a random increment to the price is chosen from a normal distribution with 0 mean and very small standard deviation. After the price is changed, the revenue earned in the next interval is

[5] Though many other algorithms were presented in their work, the market assumptions for the other algorithms did not match the market scenario of our work. For instance, we assume a monopolistic market and that there are constraints on the distribution resources. Their work was for a competitive market with no constraints on the delivery mechanism.

monitored. If the revenue earned per request is lower than before, the old price is restored. In addition, with a very small probability (called big jump probability), a new price is chosen at random. The big jump probability is much smaller than the small jump probability.

The Derivative-Following-Pricing (DFP) algorithm is similar to the TEP algorithm. An initial price in the range $[L, H]$ is chosen at random. At regular intervals, the price is varied by a random step size. If in the next interval, the revenue per customer increases, then the next increment is chosen in the same direction, i.e., price is increased. If however, the revenue decreases, then the direction of increment is reversed, i.e., the price is decreased. At all times, the price is kept in the range $[L, H]$.

4 Simulations

We performed simulations to evaluate our pricing algorithm. We implemented to model a content delivery system. All our simulations are averaged over five runs with different seed values for the random number generator. We describe the components of our simulation below.

System Description: We performed simulations with two different systems, one with a T3 (45 Mbps) outgoing link and the other with OC3 (155 Mbps) outgoing link. We chose these two link capacities to represent bandwidth capacities of a typical content provider. We assumed that there are enough servers to accommodate all the incoming requests. In this case, bandwidth is the bottleneck resource. We ran simulations on these systems to examine how the performance of the pricing algorithms varies with resource availability. In our system, customers could choose from one of two LoS— 56kbps or 128kbps. We chose request service times from a uniform distribution between 90 and 110 minutes. This closely models the typical length of movies in a VoD system[6].

Customer Choice of Products: In all our simulations we assume that there are 100 products for the customer to choose from. Customer choice of the products was assumed to follow a Zipf-like distribution with zipf-exponent[7], $\theta = 0.73$. In a Zipf-like distribution, the i^{th} popular product in a group of m products is requested with probability $\frac{\frac{1}{i^\theta}}{\sum_{j=1}^{m} \frac{1}{j^\theta}}$.

Customer Valuation Model: We assume that the products are partitioned into classes. The valuation of a product is drawn from a probability distribution which is common for all products in a class. We further assume that the content provider knows how the products have been partitioned, but has no knowledge about the probability distribution. This is a reasonable assumption because, in real-life, the content provider can partition products into "New", and "Old" classes. The valuations for products in one class can be expected to be significantly different from those of products in other classes. In our simulations, we chose the number of classes (say k) and a product was equally likely to belong to any of these k classes.

[6] Video-on-demand using a 56 kbps modem may sound unrealistic. However, one can think of downloading mp3 songs in about 100 minutes using a 56 kbps modem.

[7] Web-page accesses have been observed to obey a Zipf-like distribution with zipf-exponent in the range 0.64 to 0.83 [11].

Since humans typically think in terms of discrete values[8], we chose three possible discrete probability distributions for modelling customer valuations– Uniform, Bipolar, and Zipf. We also chose one continuous distribution– Normal. We briefly describe each of them below:

- **Uniform(l, h, n):** Customer valuations are drawn from n equally spaced values in the range l to h (both inclusive) with equal probability.
- **Bipolar(l, h, r):** Customer valuations are either l with a probability r, or h with a probability $(1 - r)$.
- **Zipf(l, h, n, θ):** Customer valuations are drawn from a set of n equally spaced values in the range l to h (both inclusive) whose ranks follow Zipf distribution with zipf-exponent θ. Income distributions are believed to correspond to a Zipf distribution with $\theta = 0.5$ [12].
- **Normal(μ,σ):** Customer valuations are drawn from a normal distribution with mean μ and standard deviation σ. We ignore negative values drawn from this distribution.

In all our simulations, our unit of currency is dimes (10 dimes = \$1). We performed simulations with numerous customer valuations. In this work, we present results for customer valuations that are "realistic". We use valuations corresponding to prices charged in movie theaters. We have observed theaters charging anywhere in the range of \$2.50 to \$8.50 for movies. We chose two classes of products. For simplicity of labelling, we shall refer to these classes as "Old", and "New". For the Uniform distribution, and 56kbps LoS, valuations were in the range $[15, 25]$ and $[25, 45]$ dimes for Old and New movies respectively. For 128 kbps LoS, valuations were in the range $[50, 70]$ and $[60, 90]$ dimes respectively. In case of the Zipf distribution, and 56 kbps LoS, valuations were in the range $[20, 35]$ and $[30, 45]$ dimes for Old and New movies respectively. For 128 kbps LoS, valuations were in the range $[45, 60]$ and $[70, 99]$ dimes respectively. In case of the Bipolar distribution, and 56 kbps LoS, valuations drawn from {15, 32}, {30, 38} dimes respectively. For the 128 kbps LoS, valuations were drawn from {50, 67} and {70, 87} dimes respectively. In case of the normal distribution, the $\langle \mu, \sigma \rangle$ of the distributions were: $\langle 20, 5 \rangle$ for (Old, 56 kbps LoS), $\langle 35, 5 \rangle$ for (New, 56 kbps LoS), $\langle 50, 10 \rangle$ for (Old, 128 kbps LoS), and $\langle 80, 10 \rangle$ for (New, 128 kbps LoS).

Pricing Policy: We assume that the content-provider will charge at least \$1 and not more than \$10 for serving the content. We simulated all three pricing algorithms described in the previous section. For the Trial-and-Error-Pricing algorithm, we set small jump probability to be 0.05 and big jump probability to be 0.001 as mentioned by Sairamesh and Kephart [8]. For the Derivative-Following-Pricing algorithm, price increments were chosen from a uniform distribution in the range $[0, 10]$. In case of the HYBRID algorithm, we chose a server load threshold of 0.75. When current server load exceeded this threshold, exponentially increasing prices were charged.

[8] In real life, customer valuations may not conform to any of these distributions. But in the absence of real life data, our objective was to test the robustness of the pricing algorithms over a range of "feasible" customer behavior patterns.

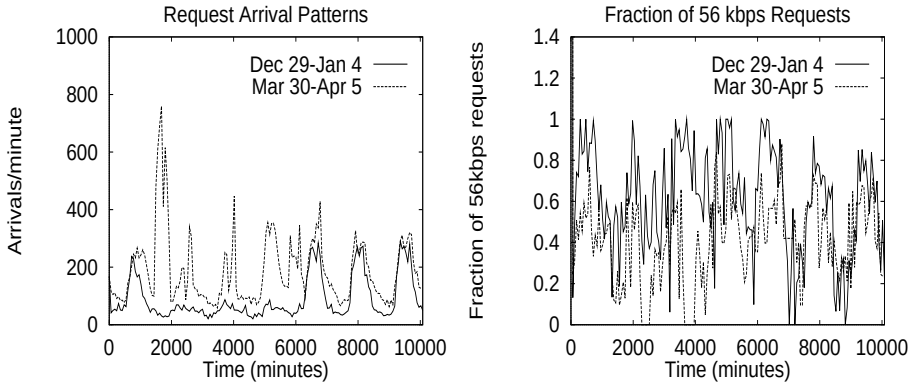

Fig. 2. Request Arrival Process

Request Arrival Process: We obtained hourly logs from a Content Delivery Network[9] for all content requests between the dates December 28, 2000 and September 8, 2001. The data we obtained consisted of the number of connections and the number of bits transmitted per second during each hour of the observation period. The data was collected over streaming servers across the United States. We obtained data for Real, Windows Media and Quicktime connections. Of these, we could only use data from Quicktime connections because of differences in the way Real and Windows Media count the number of connections. Assuming that requests could only be of two types– 56 kbps or 128 kbps, we estimated the relative fraction of requests for each level of service. We then scaled the data we obtained so that it matched the total bits transmitted by Real, Windows Media and Quicktime put together. We chose two representative weeks–December 29 to January 4, and March 30 to April 5 for running our simulations. The first week is during peak holiday season. The second week is during a normal week during the year. The request arrival rates and the fraction of 56 kbps requests for both the weeks is shown in Figure 2. Notice that the request arrivals are lower for the peak holiday season. This could possibly be because the CDN usage could have grown over the five months in between. Also, notice the the fraction of requests for 56 kbps service varies a lot in both the weeks. Though it would be tempting to draw inferences on the nature of the traffic, the growth of broadband, and other statistics based on this data, we would like to emphasize that we obtained the data based on a number of assumptions which may not hold in reality. However, these assumptions do not defeat the purpose of generating semi-realistic test cases for simulation in the absence of actual trace data.

Metrics: We use two metrics in our simulations: (1) revenue earned, and (2) percentage of requests denied service because they could not be scheduled due to lack of resources. The higher the revenue earned by a pricing algorithm, the better the performance. Ideally, we would like to compare the revenues earned by each algorithm with the predicted maximum expectation of revenue, computed using complete knowledge of system and customer parameters. However, as we

[9] We have withheld the name of the CDN upon request.

shown in our earlier work [5], the revenue maximization problem is intractable when customer valuations conform to discrete probability distributions. Even for the normal distribution, in a system with 100 products and a variety of request arrival rates, it is difficult to compute the globally optimal revenue. We therefore only use comparison among revenues earned by the different algorithms in different scenarios to characterize them. Thus, it is quite likely that even though one algorithm performs very well, the revenue earned using that algorithm may be very far from the global optimum. Our focus therefore has been to ascertain if one of the algorithms performs *consistently* well in comparison to the other algorithms across the different customer valuation and system load profiles. Our other metric, the fraction of denied requests, in very important for a commercial system for two reasons. First, a high percentage of denied requests indicates that the content-provider is not living up to service guarantees. Second, it indicates that the content provider is unable to manage available resources efficiently[10].

5 Results

We now present our simulation results. We simulated two situations:

1. There is no change in the customer behavior. However, the request arrivals are dynamic.
2. There is a change in the customer behavior during peak hours. This change corresponded to the time in the request arrival patterns when there was a sudden surge in arrivals.

To better illustrate the performance of each pricing algorithms, across different system and customer profiles, we present the results in tabular form. Each entry in the table is an ordered pair $\langle\ \mathcal{R},\ r\ \rangle$. $\mathcal{R}$ is the mean revenue earned by that pricing algorithm over a number of simulations and r is the mean request denial rate. By denial rate we mean the fraction of requests that could not be served (due to lack of resources) even though the customer agreed to the price. Note that it will not be appropriate to compare revenues within a column because, the customer valuations are different for different distributions.

Simulation 1: Since the performance was consistent through all the days of the week, we only present results for single days during the chosen weeks. We presents for Dec 29 2000 (Friday, holiday season), March 31 2001 (weekend) and April 3 2001 (weekday). Table 3 presents the results for the first set of simulations. The revenues have been rounded to the nearest 1000. In Table 3 we observe that on average, the HYBRID algorithm earns 75% to 150% more than TEP and 10% to 87% more than DFP across all the workloads and for both the system capacities. However, there were some simulations in which the TEP algorithm earned comparable revenues. This happened when the initial price chosen was close to the ideal fixed price. Even in those simulations, the HYBRID algorithm earned as much or slightly more revenue. The DFP algorithm outperforms the TEP algorithm in all the cases shown here, mainly because it learns the customer behavior better than TEP. We also observe that the service denial rate is very

[10] We make a distinction between customers who are denied service because they do not accept the price and those who accept the price but are denied service due to resource constraints.

December 29 2000

	T3 link			OC3 link		
	HYBRID	TEP	DFP	HYBRID	TEP	DFP
Uniform	⟨ 469, 0.03 ⟩	⟨ 268, 0.49 ⟩	⟨ 356, 0.69 ⟩	⟨ 1273, 0.00 ⟩	⟨ 846, 0.35 ⟩	⟨ 1151, 0.53 ⟩
Normal	⟨ 472, 0.02 ⟩	⟨ 229, 0.39 ⟩	⟨ 300, 0.62 ⟩	⟨ 1207, 0.00 ⟩	⟨ 713, 0.26 ⟩	⟨ 967, 0.47 ⟩
Bipolar	⟨ 458, 0.05 ⟩	⟨ 250, 0.49 ⟩	⟨ 388, 0.39 ⟩	⟨ 1399, 0.01 ⟩	⟨ 790, 0.35 ⟩	⟨ 1220, 0.25 ⟩
Zipf	⟨ 310, 0.01 ⟩	⟨ 130, 0.33 ⟩	⟨ 268, 0.53 ⟩	⟨ 947, 0.00 ⟩	⟨ 407, 0.21 ⟩	⟨ 866, 0.40 ⟩

March 31 2001

	T3 link			OC3 link		
	HYBRID	TEP	DFP	HYBRID	TEP	DFP
Uniform	⟨ 454, 0.03 ⟩	⟨ 219, 0.55 ⟩	⟨ 310, 0.70 ⟩	⟨ 1406, 0.01 ⟩	⟨ 652, 0.48 ⟩	⟨ 966, 0.62 ⟩
Normal	⟨ 485, 0.04 ⟩	⟨ 183, 0.50 ⟩	⟨ 260, 0.71 ⟩	⟨ 1461, 0.02 ⟩	⟨ 559, 0.43 ⟩	⟨ 850, 0.63 ⟩
Bipolar	⟨ 440, 0.04 ⟩	⟨ 208, 0.54 ⟩	⟨ 306, 0.60 ⟩	⟨ 1372, 0.02 ⟩	⟨ 627, 0.46 ⟩	⟨ 979, 0.52 ⟩
Zipf	⟨ 319, 0.00 ⟩	⟨ 125, 0.46 ⟩	⟨ 231, 0.55 ⟩	⟨ 1025, 0.00 ⟩	⟨ 416, 0.39 ⟩	⟨ 781, 0.48 ⟩

April 3 2001

	T3 link			OC3 link		
	HYBRID	TEP	DFP	HYBRID	TEP	DFP
Uniform	⟨ 428, 0.02 ⟩	⟨ 179, 0.54 ⟩	⟨ 265, 0.65 ⟩	⟨ 1264, 0.01 ⟩	⟨ 550, 0.40 ⟩	⟨ 852, 0.51 ⟩
Normal	⟨ 459, 0.04 ⟩	⟨ 163, 0.50 ⟩	⟨ 257, 0.60 ⟩	⟨ 1331, 0.01 ⟩	⟨ 513, 0.37 ⟩	⟨ 862, 0.47 ⟩
Bipolar	⟨ 435, 0.04 ⟩	⟨ 171, 0.52 ⟩	⟨ 278, 0.59 ⟩	⟨ 1321, 0.01 ⟩	⟨ 533, 0.39 ⟩	⟨ 895, 0.45 ⟩
Zipf	⟨ 325, 0.03 ⟩	⟨ 119, 0.46 ⟩	⟨ 229, 0.57 ⟩	⟨ 1033, 0.00 ⟩	⟨ 400, 0.34 ⟩	⟨ 758, 0.44 ⟩

Fig. 3. ⟨Revenue, denial-rate⟩ of Pricing Algorithms with No Changes in Customer Behavior

high for both TEP (0.21 to 0.55) and DFP (0.25 to 0.70) algorithms. This is because, they charge a low price and cannot accommodate all the requests. Such high service denial rates would be unacceptable in a commercial content delivery system. We also note that the revenues with OC3 link are higher than with the T3 link. This is clearly because the system can accommodate more requests. Note that the revenues during the weekend (March 31) are in general more than the revenues during the weekday (April 3) for all the algorithms. This is because of the differences in the request arrival pattern.

Simulation 2: In the second set of simulations, we varied the customer valuation during peak hours. For the results presented in this paper, all the customer valuations were increased by 10-20 dimes during peak hours. Table 2 presents the results. Since the results are similar to the first set, we present results only for March 31 due to space constraints. As before, the revenues have been rounded to

	T3 link			OC3 link		
	HYBRID	TEP	DFP	HYBRID	TEP	DFP
Uniform	⟨ 473, 0.04 ⟩	⟨ 240, 0.60 ⟩	⟨ 318, 0.73 ⟩	⟨ 1469, 0.02 ⟩	⟨ 711, 0.52 ⟩	⟨ 995, 0.65 ⟩
Normal	⟨ 497, 0.06 ⟩	⟨ 201, 0.55 ⟩	⟨ 270, 0.74 ⟩	⟨ 1533, 0.03 ⟩	⟨ 617, 0.48 ⟩	⟨ 873, 0.66 ⟩
Bipolar	⟨ 458, 0.05 ⟩	⟨ 242, 0.60 ⟩	⟨ 314, 0.65 ⟩	⟨ 1424, 0.03 ⟩	⟨ 715, 0.52 ⟩	⟨ 1006, 0.57 ⟩
Zipf	⟨ 340, 0.02 ⟩	⟨ 158, 0.53 ⟩	⟨ 253, 0.63 ⟩	⟨ 1097, 0.01 ⟩	⟨ 509, 0.45 ⟩	⟨ 839, 0.55 ⟩

Table 2. ⟨Revenue, denial-rate⟩ of Pricing Algorithms with Changes in Customer Behavior

the nearest 1000. The HYBRID algorithm consistently generates high revenues in comparison to the other algorithms across all customer distributions and resource constraints, mainly because it learns the customer behavior by experimenting with different prices. All the results appear consistently similar to the results in the first set of simulations. All revenues are marginally higher than in the first set because the customer valuations are higher during the peak hours. The revenues are however not significantly higher because the customer valuations did not increase significantly and moreover both the systems did not have enough capacity to satisfy all the requests. We also observe that the service denial rate is higher for all the algorithms. This is because, customers have more money to spend during peak hours, and therefore accept the quoted price more often. Note that the increase in service denial rate is higher in case of DFP (around 0.03 to 0.08) and TEP (0.04 to 0.07) than in case of HYBRID (around 0.01 to 0.02).

The reason why DFP and TEP do not perform well is that the algorithms do no consider resource constraints. They were primarily designed for a scenario where e-content can be delivered at leisure. We also ran other simulations to evaluate our choice of parameters for the algorithms. We only present a summary of our findings due to reasons of space. We observed that performance of the TEP and DFP did not vary when we changed the interval after which prices are reassessed. This was because the jumping probabilities for TEP are very small, and in case of DFP, the algorithm itself is independent of the interval. In case of the HYBRID algorithm however, we observed that a small interval generates higher revenue but increases service denial rate. We found that an interval of 45 minutes was ideal in terms of revenue as well as service denial rate. The results presented in this paper used a 45 minute interval for all the algorithms. We also observed that by increasing the big jump and small jump probability, the performace of the TEP was more erratic. The revenue did not increase or decrease in a consistent way with increasing jumping probability.

6 Conclusions

In this paper we developed an approach for pricing delivery of e-content in a system with multiple LoS. The pricing scheme, called HYBRID, was dynamic because the price varied with time. The pricing scheme was based on oberving customer reactions to price and provisioning of resources among the different levels of service. Resources were dynamically provisioned based on the amount

of resources that requests for each LoS could consume. We compared the performance of this scheme with two other simplistic dynamic pricing schemes adapted from work by other researchers. We performed simulations using semi-realistic data to evaluate the performance of the algorithms. We observed that the HYBIRD pricing scheme consistently generates high revenues across a range of customer and system profiles. We also observed that the HYBRID pricing scheme reduced the number of customers rejected service due to resource constraints mainly by charging high prices at times of peak load. We observed that the two other dynamic pricing schemes failed to generate higher revenues mainly because they do not consider resource restrictions for content delivery.

In this work, we assumed that the customer's choice of LoS is independent of the price of the content. Such an assumption is valid in today's Internet where there is a big difference in quality between content avaiable at dial-up speeds and content available at broadband speeds. Such an assumption may not be valid in the future where more customers will have broadband connectivity, and content providers will possibly provide content at a range of quality levels, each marginally different from the other. Pricing mechanisms for such markets is an avenue for future research.

References

1. S. Jagannathan and K. C. Almeroth, "Price issues in delivering e-content on-demand," *ACM Sigecom Exchanges*, vol. 3, May 2002.
2. S. Jagannathan, J. Nayak, K. Almeroth, and M. Hofmann, "E-content pricing: Analysis and simulation," tech. rep., University of California Santa Barbara, November 2001. available at http://www.nmsl.cs.ucsb.edu/papers/ECONTENTPRC.ps.gz.
3. S. Jagannathan and K. C. Almeroth, "An adaptive pricing scheme for content delivery systems," in *Global Internet Symposium*, (San Antonio, Texas, USA), November 2001.
4. S. Jagannathan and K. C. Almeroth, "The dynamics of price, revenue and system utilization," in *Management of Multimedia Networks and Services*, (Chicago, Illinois, USA), October 2001.
5. S. Jagannathan, J. Nayak, K. Almeroth, and M. Hofmann, "On pricing algorithms for batched content delivery systems," tech. rep., University of California Santa Barbara, 2002. available at http://www.nmsl.cs.ucsb.edu/papers/BatchingPrc.ps.gz.
6. P. Basu and T. Little, "Pricing considerations in video-on-demand systems," in *ACM Multimedia Conference*, November 2000.
7. J. Mackie-Mason, C. H. Brooks, R. Das, J. O. Kephart, R. S. Gazzale, and E. Durfee, "Information bundling in a dynamic environment," in *Proceedings of the IJCAI-01 Workshop on Economic Agents, Models, and Mechanisms*, August 2001.
8. J. Sairamesh and J. Kephart, "Price dynamics of vertically differentiated information markets," in *International Conference on Information and Computation Economies*, 1998.
9. J. Wolf, M. Squillante and P. Yu, "Scheduling algorithms for broadcast delivery of digital products," *IEEE Transactions on Knowledge and Data Engineering*, 2000.

10. S. Chan and F. Tobagi, "On achieving profit in providing near video-on-demand services," in *Proceedings of the 1999 IEEE International Conference on Communications (ICC'99)*, June 1999.
11. L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: Evidence and implications," in *Infocom*, pp. 126–134, 1999.
12. G. Zipf, *Human Behavior and the Principle of Least Effort, an Introduction to Human Ecology.* Addison-Wesley, 1949.