CUSTOMER BEHAVIOR MODEL FOR QUALITY-OF-SERVICE ENVIRONMENTS WITH MANY SERVICE LEVELS

Ilya Gluhovsky
Sun Microsystems Laboratories
18 Network Circle MPK18-122
Menlo Park, CA 94025
ilya.gluhovsky@sun.com

ABSTRACT

In a general on-line service environment, a service provider (SP) offers a set of service levels to customers at various prices. Higher priced service levels typically include a guarantee for faster response/completion times. The problem that an SP faces is pricing different service levels. Such a price schedule greatly impacts the *rate* of incoming requests, the *mix* of their service levels, and consequently the revenue of the SP. In this work we propose a data driven method for estimating the rate and the mix for any given price schedule. The data exclusively consist of the observed rate of request submissions and the actual choices that customers have made when presented with previously used price schedule(s). Given this characterization, the SP is able to provision resources and optimize scheduling for a given price schedule and ultimately choose the price schedule that optimizes the revenue. The problem is motivated by network service and utility computing environment applications.

Keywords: customer demand, grid computing, internet economics, machine learning, online services, probabilistic algorithms, revenue optimization, utility computing.

1. Introduction

In a general service environment, individual clients or client businesses contract with a service provider (SP) to gain access to the provider's on-line service. A client is often provided with a service level agreement (SLA), which in its basic form stipulates the payment to the SP per job unit for beginning the service of the client's request within a certain time of its arrival and the penalty that the SP pays to the client otherwise. An SLA can alternatively stipulate the response time of the system. Such general framework stands for many scenarios, such as the following:

a). *Providing content (e.g. data base access, on-line financial information) or access to a computer program ("applications on tap"), where different service levels correspond to different bandwidth requirements [Paschialidis and Tsitsiklis 2000].*
b). Voice or video connections [Paschialidis and Tsitsiklis 2000].
c). Hosting e-commerce web sites of client businesses. A client business provides an e-commerce application and the SP maintains the commercial database and services customers of the client business. The SLAs stipulate the responses for commercial transactions that originate from a customer. Liu et al. [2001] discuss this scenario in detail.
d). A high productivity computing (HPC) customer submits a large, typically multithreaded job and receives a guarantee of getting results by a certain time. In this case an estimate of the execution time is required and the time is usually stated in units of the execution time. The job unit is usually the CPU time, while other job requirements, such as storage, are not priced according to the service level.

From now on we will use the term *job* to refer to a generic service that a customer receives. Furthermore, we define a *price curve* to be a function that maps a service level of a job into its price per job unit. We assume that the price curve is set by the SP with the goal of revenue maximization. Changes to the price curve greatly impact both the job arrival rate and the service level distribution. For example, if the SP raises the price of a premium service, some customers who depend on it will leave and subscribe to a service with a competitor or maintain their own system. Therefore, the rate of arriving jobs will decrease. In addition, some customers would choose a lower service level, which is now relatively more attractive. Thus, the service level distribution would give more weight to lower service levels. We make an important observation that these rate and distribution are functions of the entire price curve.

The purpose of this work is to build a *customer behavior model* (CBM) which summarizes the choices that a typical customer makes when presented with any given price curve. That is, it gives estimates of the job arrival rate and the service level distribution for any price curve that might be contemplated by the SP. From now on, the notion of a customer is viewed broadly to also include different job or transaction types even if they originate from the same physical customer because different jobs may carry different requirements. It is clearly conceivable for the same physical customer to have some jobs that are more urgent than others.

We view a CBM as an integral part of a SP business framework. An SP is ultimately interested in maximizing his revenue. By having a CBM available, an SP knows the demand structure for any price curve that can be offered. Thus, for any given price curve the SP can accurately provision computational resources necessary to fulfill the majority of the SLAs as well as optimize job scheduling. For example, in the case of grid computing, using a performance model of a grid, such as Berman et al. [2003], an SP is able to estimate the number of machines required to service average and peak demands, determine whether an upgrade to a more powerful hardware is warranted, assess the benefit of advanced scheduling techniques, virtualization, and so forth. Moreover, an SP is able to anticipate the effect that certain changes to the price curve will have on the revenue as well as on the requirements for computing resources. Leaving the computing resources alone for a moment, an SP may optimize the revenue with respect to the price curve given the current resources and scheduling. Finally, both the price curve and the computing needs may be optimized together to achieve the largest revenue possible given the current market conditions. Additionally, we will also see that the proposed CBM also adapts naturally to changing market conditions, so that the price curve and possibly other operational parameters that are easily changeable on the fly may be adjusted in real time.

The proposed method for CBM estimation is based solely on observing customer choices over time and does not rely, for example, on any customer survey information. The methodology does not impose any operational constraints due to model training. An SP is free to choose any operational price curve or curves over time. As customers make service choices, the data are collected and are used to train CBM. No additional work on the part of an SP is necessary. The training data set comprises triples of the customer id, the chosen service level, and the price curve offered at the time.

Since a CBM constitutes a crucial component of an SP operating framework, much of the research in the latter area, Cocchi et al. [1993], Paschialidis and Tsitsiklis [2000], Low and Varaiya [1993], Paschialidis and Liu [2002], Wang et al. [1997], Wang and Schulzrinne [2006] and references therein among many others, uses some form of a CBM. CBMs are usually obtained in a heuristic manner or are just assumed to be available as convenient closed form expressions without any validation of matching observed customer behavior or giving any guidance as to how a CBM might be estimated from the latter. Thus, the present article complements much of the current research by closing the gap between the assumed and the observed customer behavior, in that CBMs obtained using the proposed approach may be used in place of those customer behavior heuristics.

In this article we follow the pioneering paper Cocchi et al. [1993] by directly expressing user utilities and prices as functions of the service level and follow Paschialidis and Tsitsiklis [2000] and Low and Varaiya [1993] by explicitly considering customers' reaction to price changes. The latter two articles take on the problems of revenue as well as welfare maximization and consider both static and dynamic price curves, the latter being dependent on the realtime state of system load. Both assume that a customer is locked in a certain class of service, where his/her demand depends on the price associated with that particular class only. As we have noted earlier, a realistic customer choice depends on the entire price curve and this is the way it is modeled here. We adopt a general multilevel model with discrete or continuous service levels and characterize multivariate relationships among them as dependent on the entire price curve. In Paschialidis and Liu [2002], a follow-up work to Paschialidis and Tsitsiklis [2000], a customer may also choose a level of service in the following restricted way. Upon being classified into a particular service level, the customer remains there as long as the utility of that service exceeds its prevailing price. In this work we assume a much more realistic scenario of choosing the level with the (near-) best utility-price differential as established in Cocchi et al. [1993]. All three articles Paschialidis and Tsitsiklis [2000], Low and Varaiya [1993], and Paschialidis and Liu [2002] rely on the availability of user demand characteristics. Thus, we make the following two contributions: we propose a much more realistic CBM and we develop its estimation methodology.

A brute force approach to estimating customer demand for a particular level of service is to fit a regression model to the observed demand as a response and the corresponding prices for *all* service levels as predictors [Fulp and Reeves 2004, Wolski et al. 2001]. To this end, for a particular service level, one collects data pairs (x_i, y_i), where y_i is the observed job arrival rate for that level during observation period i and x_i is the vector of prices for all service levels at the time. One then fits a relationship that best explains y as a function of x.

We now compare this regression approach and the proposed method. It is clear that the regression approach does not admit continuous service levels. For n (discrete) service levels under consideration, n regression models must be fitted with n predictors in each one. The n models are fitted separately. By contrast, we construct a single model for all service levels which captures multivariate interactions among them and uses data much more efficiently. Specifically, the regression approach requires that either a particular parametric functional form for each model be supplied as done in Fulp and Reeves [2004] and Wolski et al. [2001] or the models be fitted nonparametrically. In both cases one needs a large data set to obtain models with reasonable accuracy. This is because the n predictors are expected to interact in nontrivial ways, and one has to include interaction terms into the model. Wolski et al. [2001] resort to 17 degree polynomials to capture the behavior. Thus, unless the number of service levels is very small, such models are unlikely to be useful. By contrast, in Section 3 we illustrate that one can obtain very tight fits with our approach for only 1500 customer choices. Moreover, we do so for a continuous service level model. In the case of prescribing a particular functional form, one adds a fair amount of subjectivity into the analysis, while even more data are necessary to produce useful nonparametric fits. In our formulation, useful results can be obtained with very small amounts of data by using prior information as to reasonable customer behavior, which may be *noninformative*, and gradually refining the model with every new data point. This guarantees that our model is " stable" for *all* price curves as each subsequent model is at least as good as the previous one. By contrast, the data must be presented up front in the regression approach *and* must cover a wide enough n-dimensional region to include *all* candidate future price schedules, as extrapolation beyond the convex hull of the data is typically useless except for special situations [Gluhovsky and Vengerov 2007]. In addition, when many service levels are used, the fact that a customer has chosen a particular level typically means that a customer had other choices that were almost as attractive. Also, customers are not expected to choose the absolute best among the offerings all the time, but rather a satisfactory near-best. The regression model does not take advantage of this " geometry" by giving some benefit to service levels in proximity of the chosen one, while our technique does. Another benefit of our approach is that it is adaptable to new service level offering bundles (e.g. adding another level of service to the existing ones) with no additional work (based on the existing data). By contrast, a regression model (e.g. $n+1$ demand functions of $n+1$ predictors) has to be refitted from scratch and in particular, data for the old environment cannot be reused.

Wang and Schulzrinne [2006] consider a pricing problem for network bandwidth allocation. The article allows for different job types, but assumes that within each job type all users have the same given utility function, the origin of which is not established. Instead of our competitive setup (users may leave without receiving the service for the SP), a user is assumed to have a particular budget and is willing to spend it all on the service as long as the marginal utility is greater than the marginal cost. Thus, we consider a much more general case of customer behavior.

Although the framework for a revenue maximizing service provider (SP) has been thoroughly studied in the context of on-line network services, it has been explored little in utility computing, even though in some ways the latter is a special case of the former. A lot of utility computing literature focuses on job scheduling and resource allocation. In He et al. [2004] the performance is optimized with respect to a generic resource, for example, a processor count. The performance metric is a weighted average of response time and machine usage as opposed to the provider's revenue. Cirne [2002] schedules moldable jobs, those that can be parallelized in a variable number of ways, where the processor count is the resource. Similarly, Bennani and Menasce [2005] optimize response times and throughput, where the server count is the resource. Reed and Mendes [2005] reschedules the application when its execution pattern begins to deviate from patterns previously observed in a controlled environment as measured by application code instrumentation. Another notable scheduling contribution is presented in Berman et al. [2003] and references therein. We note that none of this work takes different customer preferences into account. Since an SP can in general increase revenue by charging more those customers to whom premium service matters, we propose that scheduling and resource provisioning be based on a CBM rather than on a uniform treatment of customers.

In Liu et al. [2001] profit maximization is carried out with respect to resource allocation while the prices corresponding to different service levels are taken as given. Here a CBM should be used to generate inputs for optimal allocation and there should be a feedback loop for price curve optimization. The setting in Zhang and Ardagna [2004] is very similar, with the scheduler also having a capability to turn servers on and off to save power.

Wolski et al. [2001] find the equilibrium prices for different utility computing market commodities (CPU time, memory usage, etc). Some strong simplifying assumptions are made on the behavior of providers (they sell all available units as soon as the unit price matches the historical unit price) and clients (assumed to have a fixed budget and submit as many jobs as the budget would fit for any given price). In its present form their framework does not leave room for different service levels or e-commerce business customers.

2. Customer Behavior Model

A customer faces a tradeoff between faster service and a higher cost at which it comes. Here d is an abstract notion of a delay and p is the abstract cost to the customer for choosing a service level associated with delay d. The abstract notions are introduced to keep the model as general as possible. Here we only require that the customer prefer a smaller cost and a service level associated with a smaller delay.

Example. In the High Productivity Utility Computing context, let $d = t/t_e - 1$, where t is time in the system, t_e is the n-CPU job execution time measured in hours, and let p be the dollar cost per CPU-hour. The associated SLA stipulates that the customer pays $\$p = \$p(d)$ per CPU-hour, that is, $\$p \times n \times t_e$. If the delay of the job is greater than d, that is, the job does not complete in $(1+d)t_e$ hours, the provider pays back, say, $\$p/2 \times n$ for each additional delay hour.

Let $u(d)$ be the utility function from receiving the service level associated with delay d. Since additional delay units are expected to matter progressively less, $u(d)$ is assumed to be convex (and decreasing). The set of choices for a customer includes leaving without receiving a service. It is denoted by $d = d_{\_}$. Define $u_\theta(d_{\_}) = 0$ as the customer receives no benefit. The SP specifies price curve $p(d)$ for entering the SLA associated with delay d. Of course, $p(d_{\_}) = 0$. Given the price curve, the set of feasible delay-cost points is $\{(d, p(d)) : d \geq 0 \cup d = d_{\_}\}$. A rational customer chooses optimal delay $d^* = \arg\max_d u(d) - p(d)$. In particular, $d^* = d_{\_}$ if there does not exist a service delay with a positive $u(d) - p(d)$. A similar setup is used in Cocchi et al. [1993].

In the above formulation we assume that the customer gets awarded per job. For example, a retailer gets (potential) profit from client transactions. Thus, if $u(d)$ is the expected customer profit, $u(d) - p(d)$ is the expected net operational gain to be maximized. In other situations, this may not be appropriate. For example, if a utility customer is operating within a specific budget and has an unlimited number of jobs, cheaper jobs become more valuable to a customer when the whole curve $p(d)$ shifts down because the customer can now run proportionately more cheap jobs. For instance, given a budget of $600, and prices $6 and $4 for two service levels, the customer can run 100 fast jobs or 150 slow jobs. However, with prices $4 and $2 respectively, he can run 150 fast or 300 slow jobs with the latter presumably being more valuable. The above formulation may be modified as appropriate.

The aim of an SP is to choose $p(d)$ that maximizes the revenue/profit. As part of our SP framework outlined above, our goal in this paper is to infer customer behavior summarized by the utility functions. In the ideal world, one could ask customers to provide their utility functions. This is a clearly unrealistic scenario. First, the customer may be unwilling to cooperate. Second, the customer may not be able to formulate his relative preferences in terms of a utility curve. Third, preferences may change over time. Therefore, we propose to infer customer utility functions from the choices that customers make when offered some price curve(s).

Assume there exists a collection of customer utility functions $u_\theta(d)$ indexed by parameter vector θ. A random customer i arrives and makes n_i delay choices $\mathbf{d}_i = (d_{ij}, 1 \leq j \leq n_i)$ according to his preference type θ_i. Let f denote the density of the chosen delay. Assume that when faced with a price curve $p(d)$ and given that the customer chooses to receive service, the customer with utility function u_θ makes a near-optimal choice according to the following distribution

$$d \mid \theta, p(\text{-}), d \neq d_{\_} : f(d \mid \theta, p(\text{-}), d \neq d_{\_}) = \frac{1}{K(\theta, p)} G\left(\frac{u_0^{\theta,p} - (u_\theta(d) - p(d))}{\sigma} \right), \tag{1}$$

where $u_0^{\theta,p} = \max_{d \geq 0 \cup d = d_{\_}} u_\theta(d) - p(d)$ is the optimal utility gain, a nonnegative decreasing function G and a parameter σ give the extent of departure from optimality, and $K(\theta, p)$ is the normalization constant. Note that the argument $u_0^{\theta,p} - (u_\theta(d) - p(d))$ in the departure is given by the loss of utility as opposed to the nominal distance $|d - d_\theta^*|$. Here G is a nonnegative decreasing function implying that the customer is unlikely to choose d far from the optimum. However, we allow for some degree of nonoptimality as a customer is expected to have difficulty in comparing near-optimal alternatives and would generally depart from the optimal choice by a small margin.

To complete the definition of delay density f, we have to define the probability of leaving. Let $d_+^* = \arg\max_{d \geq 0} u_\theta(d) - p(d)$, where the maximum is only taken over the choices where service is received. We model the odds of receiving service as being proportional to the ratio of the best G-value among the available service levels and that of leaving:

$$\frac{P\{d \neq d_- \,|\, \theta, p(\text{-})\}}{P\{d = d_- \,|\, \theta, p(\text{-})\}} = \frac{G\left(\left[u_0^{\theta,p} - (u_\theta(d_+^*) - p(d_+^*))\right]/\sigma\right)}{G\left(\left[u_0^{\theta,p} - (u_\theta(d_-) - p(d_-))\right]/\sigma\right)} = \frac{G\left(\left[u_0^{\theta,p} - (u_\theta(d_+^*) - p(d_+^*))\right]/\sigma\right)}{G\left(u_0^{\theta,p}/\sigma\right)} \tag{2}$$

One is still penalized by the departure of $u_0^{\theta,p} - (u_\theta(d) - p(d))$ from the optimum scaled by G. If d_- is optimal, $u_0^{\theta,p} = 0$ and any service choice $d_1 \geq 0$ incurs the unscaled penalty of $-(u_\theta(d_1) - p(d_1)) \geq 0$, the loss it causes (by multiplying the odds of d_1 against d_+^* given by (1) and the odds of d_+^* against d_- given by (2)). If d_+^* is optimal, d_- incurs the penalty of $u_0^{\theta,p} = u_\theta(d_+^*) - p(d_+^*) \geq 0$ of passing by an opportunity of positive value.

Our goal is to learn the distribution π of θ since it characterizes customer behavior. Indeed, if we know π, we can compute the distribution of the chosen service levels for any given price curve via

$$f(d \,|\, p(\text{-})) = \int f(d \,|\, \theta, p(\text{-}))\pi(\theta)d\theta. \tag{3}$$

Assume that π comes from a family parameterized by hyperparameter τ, $\pi(\theta\,|\,\tau)$. Let $\xi(\tau)$ denote the prior distribution on τ, which summarizes our uncertainty about τ before seeing the customer data. Upon seeing N customers with observed delay vectors $\mathbf{d} = (\mathbf{d}_i, 1 \leq i \leq N)$, the posterior distribution of τ becomes

$$\xi(\tau\,|\,\mathbf{d}) \propto \xi(\tau)\int f(\mathbf{d}\,|\,\theta,\tau)\pi(\theta\,|\,\tau)d\theta = \xi(\tau)\prod_{i=1}^{N}\int \pi(\theta_i\,|\,\tau) f(\mathbf{d}_i\,|\,\theta_i)d\theta_i =$$

$$\xi(\tau)\prod_{i=1}^{N}\int \pi(\theta_i\,|\,\tau)\prod_{j=1}^{n_i}\left[\begin{array}{c}\dfrac{1}{K(\theta_i, p_{ij})} G\left(\dfrac{u_0^{\theta_i,p_{ij}} - (u_{\theta_i}(d_{ij}) - p_{ij}(d_{ij}))}{\sigma}\right)\mathbf{1}_{\{d_{ij} \neq d\_\}} + \\ P\{d_{ij} = d_- \,|\, \theta_i, p_{ij}\}\mathbf{1}_{\{d_{ij} = d\_\}}\end{array}\right] d\theta_i, \tag{4}$$

where $\mathbf{1}$ is the indicator function.

It might be tempting to suggest a simpler model by omitting τ and inferring $\pi(\theta\,|\,\mathbf{d})$. Unfortunately, this formulation is incorrect, as such a distribution tells us about the uncertainty about the value of θ and allows one to contrast different values. Here we are after a distribution on θ. Therefore, we need a model that compares different distributions, so the unknown is τ that indexes candidate distributions. To contrast these further, consider an infinite data case. Under some regularity conditions, in the proposed formulation we obtain the distribution $\pi(\theta\,|\,\tau)$ with certainty, and this is what we are after. Following the alternative approach, we would learn the value of θ with certainty, which is meaningless.

The required distribution over θ is then

$$\pi(\theta \mid \mathbf{d}) = \int \pi(\theta \mid \tau)\xi(\tau \mid \mathbf{d})d\tau \tag{4}$$

First, we have to ensure that equations (4) and (3) are computationally feasible. Second, since a CBM is to be used as part of an optimization framework, the model for $p(d)$ should allow for straightforward introduction of local changes to the curve. It is easy to see that without loss of generality the optimal $p(d)$ can be restricted to be nonincreasing, for curve $p'(d) = \min_{s \le d} p(s)$ results in the same choices for all utility curves. We typically expect $p(d)$ to be convex. To impose derivative constraints on $p(d)$ and enable local changes during the optimization step, we use a particular wavelet basis and restrict expansion coefficients. Specifically, let $\varphi(x)$ satisfy conditions set out in Lemma 1 of Anastassiou and Yu [1992]. We will use

$$\varphi(x) = \begin{cases} 0, & x \le -1.5,\ x \ge 1.5 \\ .5(1.5+x)^2, & -1.5 \le x \le -.5 \\ 1+x-(.5+x)^2, & -.5 \le x \le .5 \\ .5(1.5-x)^2, & .5 \le x \le 1.5 \end{cases} \tag{5}$$

depicted in Figure 1a. It is shown in Anastassiou and Yu [1992] that for any integer k, function

$$p(d) = \sum_{j=-\infty}^{\infty} c_j \varphi(2^k d - j) \tag{6}$$

is nonnegative nonincreasing if c_j is a nonnegative nonincreasing sequence. Since in practice only finitely many c_j are nonzero, we also note that if the support of φ is $[-a,a]$, $g(d)$ is nonnegative nonincreasing for $d \in 0,+\infty)$ if the first nonzero c_j occurs for $j \le -a$, and from that point on c_j are nonincreasing. It is also shown that if c_j is a convex sequence, i.e. increments $c_j - c_{j-1}$ are nondecreasing, then $p(d)$ is convex. We observe that by varying coefficient c_j, $p(d)$ is only changed over $[-a/2^k, a/2^k]$.

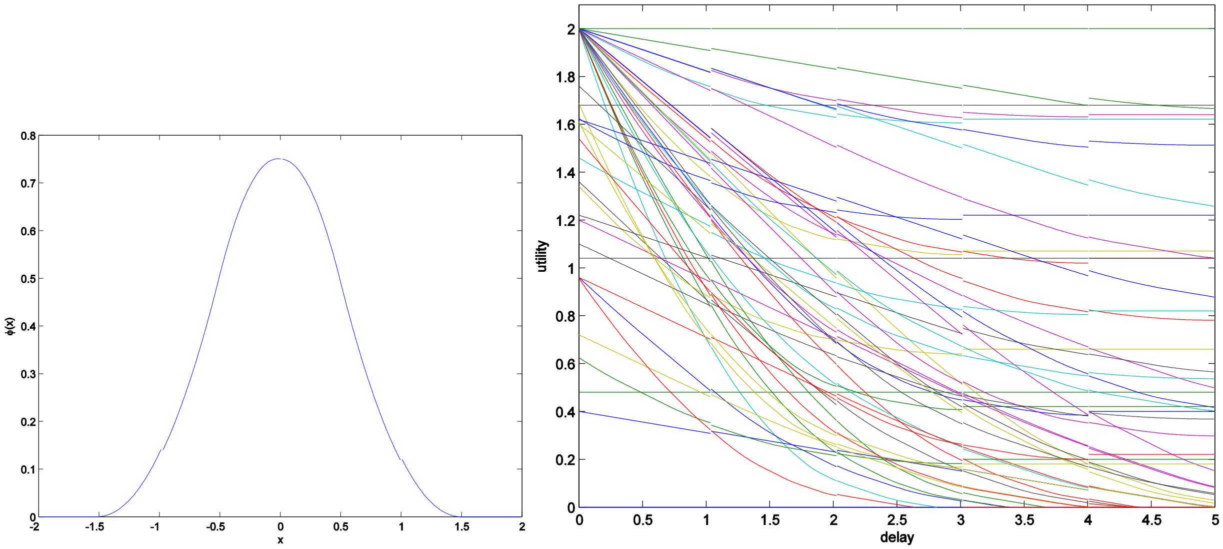

Figure 1: a. Wavelet scaling function ϕ. b. 50 utility curves u_θ.

We now describe the estimation procedure. For a fixed k assumed for simplicity to be the same as that in (6), let $u_\theta(d) = \sum_{j=-\infty}^{\infty} \theta_j \varphi(2^k d - j)$ be the utility function of customer type θ. For price curve $p(d)$

$= \sum_{j=-\infty}^{\infty} c_j \varphi(2^k d - j)$ assumed for simplicity to be defined for the same k , the density in (1) becomes

$$f(d \mid \theta, p(\text{-}), d \neq d_{\_}) = \frac{1}{K(\theta, p)} G\left(\frac{\left[\max_{d'} \sum_{j=-\infty}^{\infty} (\theta_j - c_j)\varphi(2^k d' - j) \right] - (u_\theta(d) - p(d))}{\sigma} \right). \qquad (7)$$

For the chosen φ in (5), some arithmetic shows that

$$\max_{d'} \sum_{j=-\infty}^{\infty} a_j \varphi(2^k d' - j) = \max_{j: a_j \geq a_{j-1}, a_{j+1}} \frac{3a_j^2 - a_j(a_{j+1} - a_{j-1})}{2(2a_j - a_{j+1} - a_{j-1})} \qquad (8)$$

or the maximum is on the boundary on the delay region. The complexity of (8) has two consequences. First, the normalization constant $K(\theta, p)$ is difficult to obtain as this requires numerical integration over a convex domain of vector θ . The evaluation of the integral in the right-hand side of (2) and subsequent computations for (4) is generally intractable for realistic $\pi(\theta \mid \tau)$. Since the distribution is in general not computable, a standard Monte Carlo method for drawing a sample from (2) can not be implemented. An alternative approach is to reduce the space of θ to a moderately sized representative collection $\theta_m, 1 \leq m \leq M$. Computation of normalization constants K in (2) and subsequently the integrals, which become sums, then becomes easy for any given τ and the delay distributions $f(d \mid \theta_m, p(\text{-}))$ can give interpretable insights of customer behavior.

It is important for the collection θ_m to avoid redundancy in covering the space of nonincreasing convex sequences, so that the collection is as representative as possible given its size. We use the maximum entropy experimental design methodology described in Currin et al. [1991]. The idea is to choose the M curves that fill the space uniformly. M is chosen large enough, so that no part of the space remains unexplored. We note that the local nature of the chosen wavelet representation of the utility curves allows us to substitute vector distances for curve distances, so that the methodology in Currin et al. [1991] can be used directly. The convexity constraint is imposed within the search algorithm of Currin et al. [1991] by not allowing the search paths to wander outside the nonincreasing-convexity domain. As an example, Figure 1b shows $M = 50$ utility curves for a particular choice of parameters in the algorithm of Currin et al. [1991].

Example. We consider an important case of $G(x) = \exp(-x)$. This choice of G implies that the relative odds of d_1 and d_2 only depend on the utility gain $(u_\theta(d_1) - p(d_1)) - (u_\theta(d_2) - p(d_2))$ and not on the utility level. A dollar is a dollar, so to speak. In this case, $u_0^{\theta,p}$ can be taken out of equation (1) since it enters the normalization constant and (1) becomes

$$f(d \mid \theta, p(\text{-})) \propto G\left(\frac{-(u_\theta(d) - p(d))}{\sigma} \right) = \exp\left(\frac{u_\theta(d) - p(d)}{\sigma} \right).$$

Note that $u_\theta(d)$ is bounded, so the density is proper over bounded delays. Using the conventions leading to equation (7), letting $a_j = \theta_j - c_j$, and $k = 0$ to simplify the notation, $f(d \mid \theta, p(\text{-})) \propto \exp\left(\sum_{j=-\infty}^{\infty} a_j \varphi(d - j)/\sigma \right)$. For $d \in i - .5, i + .5]$,

$$\sum_{j=-\infty}^{\infty} a_j \varphi(d - j) = (a_{i+1} + a_{i-1} - 2a_i)(d - i)^2/2 + (a_{i+1} - a_{i-1})(d - i)/2 + (6a_i + a_{i+1} + a_{i-1})/8$$

$$\overset{\square}{=} l_{i1}(d - i)^2/2 + l_{i2}(d - i) + l_{i3},$$

where l_{ik} are the linear functions in a_j. Denoting $I(x) = \int_{-\infty}^{x} \exp(s^2/2)ds$, which is not available in closed form,

$$K(\theta, p) = \sum_{i=-\infty}^{\infty} \frac{\exp\left(l_{i3} - l_{i2}^2/(2l_{i1})\right)}{\sqrt{l_{i1}}} \left[I\left(\frac{l_{i2}}{\sqrt{l_{i1}}} + \frac{\sqrt{l_{i1}}}{2}\right) - I\left(\frac{l_{i2}}{\sqrt{l_{i1}}} - \frac{\sqrt{l_{i1}}}{2}\right)\right] \mathbf{1}_{\{l_{i1} \neq 0\}}.$$

Thus, we have obtained normalized data densities (1). However, it is also clear that it is infeasible to compute the integrals in the right-hand side of (2). Therefore, even despite the exponential form of G, the implementational problems with a general scheme are likely to remain.

We now describe a model for τ. Let us begin with a moderate collection $\tau_k^{'}, 1 \le k \le K$. Let $\pi_k(\theta) = \pi(\theta \mid \tau_k^{'})$. In the examples in Section 3 we will use $K = M$ with $\pi_k(\theta)$ putting the unit mass on θ_k. Since the integrals in (2) are sums, it is easy to evaluate them for each k as noted above. Denote these by $I_k(\mathbf{d}_i)$:

$$I_k(\mathbf{d}_i) \triangleq \int \pi_k(\theta) f(\mathbf{d}_i \mid \theta) d\theta \tag{9}$$

Now consider the set of distributions over θ obtained by mixing the $\pi_k(\theta)$. Let τ stand for the mixing vector, $\pi(\theta \mid \tau) = \sum \tau_k \pi_k(\theta)$ with $\sum \tau_k = 1, \tau_k \ge 0$. Then $\xi(\tau \mid \mathbf{d}) \propto \xi(\tau) \prod_{i=1}^{N} \sum \tau_k \quad I_k(\mathbf{d}_i)$, which is a polynomial in the τ_k. The (4) becomes

$$\pi(\theta \mid \mathbf{d}) \propto \sum_l \pi_l(\theta) \int \tau_l \xi(\tau) \prod_{i=1}^{N} \sum_{k=1}^{K} \tau_k I_k(\mathbf{d}_i) d\tau \tag{10}$$

The integrand is a polynomial in the τ_k and thus, the $(K-1)$-dimensional integral can be evaluated analytically over $\sum \tau_k = 1$ provided $\xi(\tau)$ has a simple form. Here we choose $\xi(\tau)$B1. Unfortunately, the number of summands in the integral is K^N, which would typically be beyond computational capability. However, note that the integrals in (10) are means of the τ_k under $\xi(\tau \mid \mathbf{d})$ and Monte Carlo methods can be used to facilitate inference. We use Gibbs sampler [Gilks 1996] to generate a sample of $\tau^{(j)}, 1 \le j \le J$ whose limiting distribution is $\xi(\tau \mid \mathbf{d})$ by resampling one coordinate $\tau_k, 1 \le k \le K-1$ at a time in a round-robin fashion. During an update of τ_l, the new value $\tau_l^{(j+1)}$ is sampled from the Gibbs update density for $\xi(\tau_l \mid \mathbf{d}, \tau_{-l}^{(j)})$, where τ_{-l} stands for the vector of all coordinates except for the l th one. Note that $\xi(\tau_l \mid \mathbf{d}, \tau_{-l})$ is a univariate polynomial of degree N with an interval support $\left[0, 1 - \sum_{k \neq l} \tau_k\right]$. τ_K is updated after every Gibbs update via $\tau_K^{(j)} = 1 - \sum_{k=1}^{K-1} \tau_k^{(j)}$. Once the sample is computed, the integrals in (10) are estimated by

$$\int \tau_l \xi(\tau) \prod_{i=1}^{N} \sum_{k=1}^{K} \tau_k I_k(\mathbf{d}_i) d\tau \approx \frac{1}{J} \sum_{j=1}^{J} \tau_l^{(j)}, \tag{11}$$

and the evaluation of (10) is now straightforward.

The proposed procedure can be summarized as follows.

Data Collection. During the course of service operation, record the pairs of price curves offered to customers and the corresponding service level choices including leaving without receiving service. Also annotate pairs that pertain to the same customer (more precisely, the same customer-job/transaction type).

Step 1. Generate M nonincreasing convex curves to serve as a representative collection of the set of customer utility functions.

Step 2. For each customer i, compute vectors $f(\mathbf{d}_i \mid \theta_m), 1 \le m \le M$.

Step 3. Obtain $I_k(\mathbf{d}_i)$ by summing over m as in (9).

Step 4. Estimate the means of the τ_k under $\xi(\tau \mid \mathbf{d})$ via Gibbs sampler.

Step 5. Compute $\pi(\theta \mid \mathbf{d})$ via (10).

Inference. Now the service level choice distribution is given by (3) for any price curve.

We remark that the proposed procedure is related to kernel density estimation. The latter estimator, somewhat generalized, is defined by

$$\pi(\theta \mid \mathbf{d}_i, 1 \le i \le N) = \frac{1}{N}\sum_{i=1}^{N} K\left(\frac{\rho(\mathbf{d}_i, \theta)}{\sigma}\right),$$

where K is the kernel, ρ is a distance between the customer observation vector and the behavior parameter. Combining observations from the same customer and using G as before, obtain

$$\pi(\theta \mid \mathbf{d}) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{K(\mathbf{d}_i)}\prod_{j=1}^{n_i} G\left(\frac{u_0^{\theta,p_{ij}} - (u_\theta(d_{ij}) - p_{ij}(d_{ij}))}{\sigma}\right), \tag{12}$$

where $K(\mathbf{d}_i)$ is the normalization constant for the product. To compute $K(\mathbf{d}_i)$, the product must be integrated over θ. Thus, $u_0^{\theta,p_{ij}}$ remains under the integral sigh even for the exponential G and this integration is difficult in view of (8). In addition, estimator (12) is sensitive to a particular choice of a moderately sized collection of θ. As a simple example, consider two candidates θ_1 and θ_2 vs. θ_1, θ_2 and $\theta_3 \approx \theta_1$ and suppose the true $\pi(\theta)$ gives the weights of $1/2$ to θ_1 and θ_2. In the first case, the estimator works. In the second case, however, θ_1 and θ_3 receive approximately equal weights because they are close to each other and close to the weight of θ_2 because θ_1 and θ_2 are equally likely. Upon normalization, our estimate becomes approximately $(1/3, 1/3, 1/3)$, which is incorrect. The proposed procedure resolves this problem by introducing parameter τ that indexes candidate distributions of θ.

3. Results

In this section we apply the proposed methodology to customer data generated by a simulation model. A nonincreasing convex utility curve is generated at random for each customer by drawing a nonincreasing convex sequence uniformly from the unit cube and using it as wavelet basis expansion coefficients as in (6). We use a set of four price curves for training and another set of five curves for testing. These are shown in Figure 2a. Although in a real life situation drastic changes to the price curve are not expected, we allow a fair degree of disparity to illustrate the success of the methodology. Each customer makes between one and four choices with the training curves drawn at random without replacement. Thus, we have between one and four data points for each customer. We use $G(x) = \exp(-x)$ with $\sigma = .2$. Further, we carry out the experimental design procedure to generate 100 generic customer types θ that are similar to those graphed in Figure 1b. We use the collection of distributions $\tau_k^{'}, 1 \le k \le 100$ with $\tau_k^{'}(\theta) = \mathbf{1}_{\{\theta=\theta_k\}}$, which puts the unit mass on the corresponding θ_k. In the first example we have 1,500 customers.

Figure 2b depicts the cumulative distribution functions of the chosen delays corresponding to price curves 3 (in the training set), 6 and 9 (in the test set). The solid curves are our estimates while the dashed ones are those for the empirical distribution of the simulated data (not used for inference in the case of test curves 6 and 9). The vertical space at the delay of 5 between the cdf value and one is the probability that a customer leaves without receiving service. The close match within the corresponding pairs is apparent.

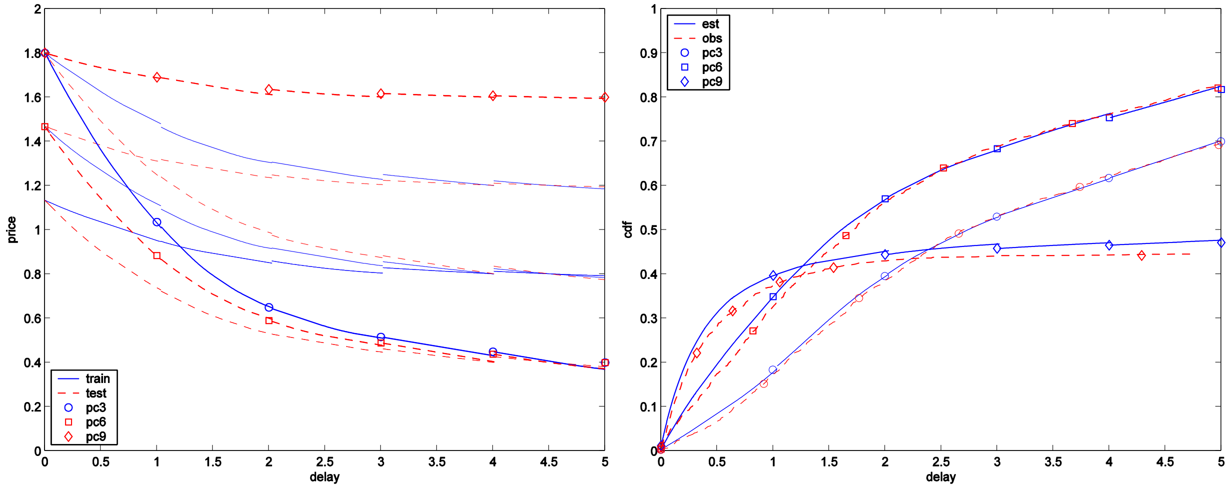

Figure 2: a. 4 price curves used for training (solid) and 5 test price curves (dashed). b. CDFs of the estimated and observed delay distributions for price curves 3, 6, and 9.

Table 1 summarizes the comparison of the estimated and the actual quantities for all 9 price curves in Figure 2a. Here we report the means and standard deviations of the 9 delay distributions given that the customer indeed receives service (the corresponding cdfs are depicted in Figure 2b for the 3 curves). In addition, we show the probabilities that a customer leaves without receiving a service. We also report the mean revenue obtained from servicing a customer assuming that the corresponding SLA is fulfilled and so no penalty is assessed. We note the close match between the estimated and actual quantities.

Table 1: Comparison of the estimates against simulation for the 9 price curves (pc1-pc4 are the training set, pc5-pc9 are the test set. Rows 1, 2: Mean and standard deviation of the chosen delay. Rows 3 and 4: The probability that a customer leaves without receiving service. Rows 5 and 6: Mean revenue from one transaction. Similar for the bottom half.

	pc1	pc2	pc3	pc4	
Delay (est)	.74, .94	1.05, 1.13	2.03, 1.34	.95, 1.08	
Delay (obs)	.74, .90	1.03, 1.08	2.04, 1.30	.90, 1.04	
P{leave} (est)	.11	.25	.30	.51	
P{leave} (obs)	.08	.22	.31	.54	
Revenue (est)	.91	.89	.57	.76	
Revenue (obs)	.93	.91	.56	.72	
	pc5	pc6	pc7	pc8	pc9
Delay (est)	1.17, 1.19	1.58, 1.31	.64, .86	1.48, 1.28	.57, .79
Delay (obs)	1.13, 1.10	1.63, 1.28	.63, .80	1.45, 1.24	.53, .71
P{leave} (est)	.08	.18	.27	.44	.53
P{leave} (obs)	.08	.18	.25	.47	.56
Revenue (est)	.72	.69	1.02	.69	.82
Revenue (obs)	.72	.68	1.04	.66	.78

In our second example we confine the study to 200 customers, but allow them to make 23 choices for 23 different price curves. This situation may arise when customers keep submitting jobs with similar requirements on their completion. The amount of data is roughly the same as that in the previous study. We generate a test set of 22 price curves, four of which are plotted in Figure 3a. All the training and test curves (including those shown) are obtained by connecting the squares shown along the vertical line at the delay of zero in Figure 3a with those at the delay of five and keeping nonincreasing curves only. The accuracy of our results is similarly high with discrepancies in the ballpark of those in Table 1. In particular, the mean revenue per transaction is 3.8% off on average across the 22 test curves. Figure 3b shows the estimated and observed delay distributions for the four test curves plotted in Figure 3a.

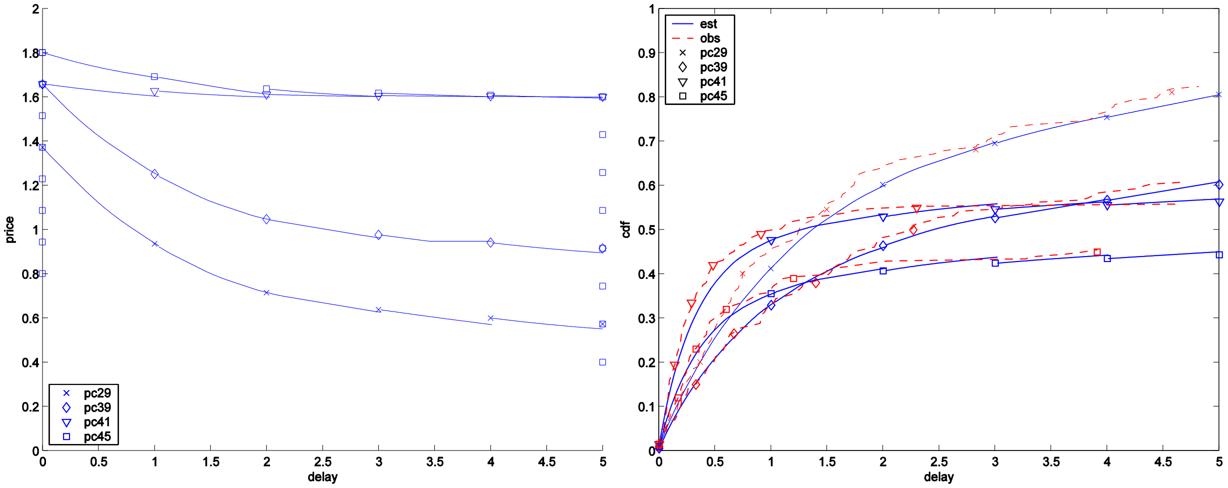

Figure 3: a. Four test price curves. b. CDFs of the estimated and observed delay distributions for these price curves.

4. Conclusion

In this work we have proposed a way to build a customer behavior model (CBM) that anticipates the job arrival rate and the service level distribution for any (new) price curve. The model is trained using only the actual choices that customers make during routine service activity. In our examples we have shown that our predictions are in close agreement with both observed and unobserved (by the procedure) simulation data over a wide variety of price curves. Since the CBM is to be used as part of a price curve optimization framework, the price curve is modeled via a particular wavelet basis to allow for easy introduction of local changes to it. The same model is used for utility curves not only to simplify computations, but also to be able to substitute vector distances for curve distances in the experimental design procedure.

There are many useful extensions to the proposed model. One such extension is required in a situation where the event of a customer leaving without receiving a service is either unobservable altogether or may take place for reasons other than there being too high prices at all service levels. We conjecture that such a complication may be alleviated by adopting a script that would invoke a pop-up question to a leaving customer to state the reason for leaving.

We have stated that our model can be easily made adaptive to changing market conditions. Indeed, more recent observations can carry greater weight by raising the corresponding data density terms in (2) to an annealing-type [Kirkpatrick et al. 1983] power greater than one. To see it by example, a power of two would be equivalent to having another identical observation.

Another useful extension has to do with updating the target distribution $\pi(\theta\,|\,\mathbf{d})$ when new data come along. Customers would hopefully provide new data points on a regular basis and recomputing the target distribution from scratch is wasteful. Instead, we can use *importance weights* [Gilks 1996] on the sample generated using Gibbs sampler to correct for the changing $\xi(\tau\,|\,\mathbf{d})$ by taking a weighted average in (11) with the weights defined as normalized ratios of the new over the old $\xi(\tau\,|\,\mathbf{d})$ evaluated at the sampled τ. Of course, the $I_k(\mathbf{d}_i)$ corresponding to the new data must be evaluated, but no additional sampling is necessary for incremental changes.

Seasonality may play an important role in setting customer preferences. For example, flower shops get most business around Valentine's Day and Mother's Day. The utility from a single transaction typically increases since the shop can charge higher prices during these periods. The rate of arrivals also increases. Payroll activity picks up at the end of each quarter and during the tax season. Large computational jobs are more likely to be submitted during the work day. At times the results are needed by next morning, but there is no utility from receiving them earlier in the middle of the night. To take seasonality into account, we would have to introduce the time variable into the utility curves as $u_\theta(d,t)$. Interchanging low- and high-pass filtering, we expect to separate seasonality effects of different periods (days, quarters, etc.) similarly to Cleveland et al. [1990]. We plan on presenting a detailed procedure in a future article. Presently, the popular solution is to split the time axis into appropriate intervals and use the analysis on each piece separately [Paschialidis and Tsitsiklis 2000, Fulp and Reeves 2004], which can be done within our methodology as well.

In this paper we have assumed for simplicity to be dealing with one particular service type for all customers. In a more realistic setting of several types of services or transactions, for example, both voice and video connections, both "browse" and "sell" transactions (with different service levels offered within each type), the proposed procedure should be repeated for different service types. For instance, an e-commerce business derives different utilities from "browse" and "sell" transactions and this should be reflected by offering different price curves.

In a future work we also plan to introduce a concept of pricing utility jobs with unusual characteristics based on opportunity cost analysis. So far we have implicitly assumed that for any arriving job we have already seen enough operationally similar jobs that we have an appropriately shaped price curve for it. But suppose, for example, that we have only seen nonpreemptible jobs, those that cannot be suspended and restarted later from roughly the same point of execution. When a preemptible job comes along, a service provider is generally able to schedule it more efficiently as should therefore charge less for it. However, since preemptible jobs are not part of the available data, price curve determination has to be accomplished differently. We believe that this can be done by determining the associated opportunity cost or, rather, revenue of not having other "conventional" jobs run in its place.

Estimation of locality parameter σ from the data is an interesting problem and parallels one in probability density estimation.

Several successive choices made by a customer are generally not independent. If the price curve stays put, over time the customer will be getting closer to his optimal choice. When faced with a completely different price curve, the new choice may be modeled as independent. However, since we do not anticipate many drastic changes to the price curve, customers may utilize their experience with previous curves. The effect on the model is that the posterior density is expected to be tighter around the optimum than what is suggested by (2). An improvement of model (2) will be an interesting future work item.

REFERENCES

Anastassiou, G.A. and X.M. Yu, ``Convex and Coconvex Probabilistic Wavelet Approximation," *Stochastic Analysis and Applications*, Vol. 10, No. 5:507-521, 1992.

Bennani, M. N. and D. A. Menasce, ``Resource Allocation for Autonomic Data Centers using Analytic Performance Models," *Proceedings of the Second International Conference on Autonomic Computing*, 229-240, 2005.

Berman, F., R. Wolski, H. Casanova, W. Cirne, H. Dail, M. Faerman, S. Figueira, J. Hayes, G. Obertelli, J. Schopf, G.

Shao, S. Smallen, N. Spring, A. Su, and D. Zagorodnov, ``Adaptive Computing on the Grid Using AppLeS," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 14, No.4:369 - 382, 2003.

Cirne, W. and F. Berman, ``A model for moldable supercomputer jobs," *Proceedings of the 15th International Parallel and Distributed Processing Symposium*, 59-79, 2001.

Cirne, W. and F. Berman, ``A comprehensive model of the supercomputer workload," *Proceedings of the 4th IEEE International Workshop on Workload Characterization*, 140-148, 2001.

Cirne, W. ``Using Moldability to Improve the Performance of Supercomputer Jobs," *Journal of Parallel and Distributed Computing*, Vol. 62:1571-1601, 2002.

Cocchi, R., S. Shenker, D. Estrin, and L. Zhang, L. ``Pricing in Computer Networks: Motivation, Formulation, and Example," *IEEE/ACM Transactions on Networking*, Vol. 1, No. 6:914-927, 1993.

Cleveland R.B., W.S. Cleveland, J.E. McRae, and I. Terpenning, ``STL: A Seasonal-Trend Decomposition Procedure

Based on Loess" (with Discussion), *Journal of Official Statistics*, Vol. 6: 3-73, 1990.

Currin, C., T.J. Mitchell, M.D. Morris, and D. Ylvisaker, ``Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments," *Journal of American Statistical Association*, Vol. 86:953--963, 1991.

Fulp, E.W. and D.S. Reeves, ``Bandwidth Provisioning and Pricing for Networks with Multiple Classes of Service," *Computer Networks Journal*, Vol. 46, No. 1:41-52, 2004.

Gilks, W.R., S. Richardson, and D.J. Spiegelhalter, Markov Chain Monte Carlo in Practice, FL, Boca Raton:Chapman and Hall, 1996.

Gluhovsky, I. and D. Vengerov, ``Constrained Multivariate Extrapolation Models with Application to Computer Cache Rates," *Technometrics*, Vol 49, No. 2:129-137, 2007.

He, L., S.A. Jarvis, D.P. Spooner, X. Chen, and G.R. Nudd, ``Performance-Based Workload Management for Multiclusters and Grids," *Proceedings of IEE-Software*, Vol. 151, No. 5:224-231, 2004.

Low, S.H and P.P. Varaiya, ``A New Approach to Service Provisioning in ATM Networks," *IEEE/ACM Transactions on Networking*, Vol. 1, No. 5:547-553, 1993.

Kirkpatrick, S., C.D. Gelatt, Jr., and M.P. Vecchi, ``Optimization by Simulated Annealing," *Science*, Vol. 220:671-680, 1983.

Liu, Z., M.S. Squillante, and J.L. Wolf, ``On Maximizing Service-Level-Agreement Profits," *Proceedings of the 3rd ACM conference on Electronic Commerce*, 213-223, 2001.

Paschialidis, I.C. and J.N. Tsitsiklis, ``Congestion-Dependent Pricing of Network Services," *IEEE/ACM Transactions on Networking*, Vol. 8, No. 2:171-184, 2000.

Paschialidis, I.C. and Y. Liu, ``Pricing in Multiservice Loss Networks: Static Pricing, Asymptotic Optimality, and Demand Substitution Effects," *IEEE/ACM Transactions on Networking*, Vol. 10, No. 3:425-438, 2002.

Reed, D.A. and C.L. Mendes, ``Intelligent Monitoring for Adaptation in Grid Applications," *Proceedings of IEEE*, Vol. 93, No. 2:426-434, 2005.

Wang, Q., J.M. Peha, and M.A. Sirbu, ``Optimal Pricing for Integrated Services Networks," *Internet Economics*, L.W.

McKnight and J.P. Bailey (eds.), MA, Cambridge:MIT Press, 353-376, 1997.

Wang, X. and H. Schulzrinne, ``Pricing Network Resources for Adaptive Applications," *IEEE/ACM Transactions on Networking*, Vol. 14, No. 3:506-519, 2006.

Wolski, R., J.S. Plank, T. Bryan, J. Brevik, ``G-commerce: Market Formulations Controlling Resource Allocation on the Computational Grid," *Proceedings of the 15th International Parallel and Distributed Processing Symposium*, 46-53, 2001.

Zhang, L. and D. Ardagna, ``SLA Based Profit Optimization in Autonomic Computing Systems," *Proceedings of the 2rd International Conference on Service Oriented Computing*, 173-182, 2004.