Time-Of-Use Pricing Policies for Offering Cloud Computing as a Service

Denis Saure, Anshul Sheopuri, Huiming Qu, Hani Jamjoom, Assaf Zeevi
Email: dsaure@gsb.columbia.edu, {sheopuri, hqu, jamjoom}@us.ibm.com, assaf@gsb.columbia.edu

***Abstract*—We study a reservation system with finite computing resources over an infinite horizon, where a set of incumbent users submit reservation requests for computing resources ahead in time. Computing resources may be purchased in exchange for tokens. We use the Multinomial Logit (MNL) framework to model customer substitution behavior. Given user requests, the objective is to maximize system performance, defined as the proportion of customers that obtain their preferred time slot, by adjusting resource prices in tokens per unit of time and per computing resource.**

We consider a class of pricing policies called Time-of-Use (ToU), and propose a simple and intuitive algorithm that is provably optimal for an approximation to our formulated problem. Our proposed solution has the appealing property of *flattening demand* over the horizon. We evaluate the performance of our approach numerically. For the set of problem instances that we consider, the optimal ToU policy outperforms single pricing strategies by 3-8 % for Customer Satisfaction, on average. We discuss the implementation of our proposed approach for Cloud Computing being developed by IBM at the King Abdullah University of Science and Technology (KAUST).

I. Introduction

Cloud Computing is offering a new paradigm for the dynamic provisioning of scalable and efficient services. Several firms, including Amazon, IBM and Google, are making large investments in Cloud Computing. Furthermore, analysts such as Gartner believe that the introduction of Cloud Computing in the market will result in strategic shift in the IT industry: "Organizations are switching from company-owned hardware and software assets to per-use service-based models" [8]. To bring this new technology to the market, firms are not only addressing the technology questions related to e-migration, virtualization, etc. but also developing new scheduling and pricing models in order to efficiently allocate computing resources on the cloud to consumers. Research in these areas has the potential to impact the evolution of this business. The focus of this paper is to address the problem of pricing of computing resources.

Consider a firm that owns a Cloud Computing center (C3) consisting of a finite number of computing resources, which we refer to as compute nodes. We assume that the firm has a pool of captive users.[1] This would be the case if the firm was a university and the users were students and faculty or if the firm were a for-profit company and the users were paying customers. In order to manage access to C3, each user is periodically endowed with a fixed number of tokens that can be exchanged for compute nodes by submitting a request through a reservation system (R), depending on the price (throughout the paper, we refer to prices in tokens) charged by the manager of C3.

We assume that each user has full visibility into the system workload. In particular, when a user interacts with R, he may observe which time slots are currently reserved and which are not, as well as posted prices for each compute node that has not been reserved. Given the number of compute nodes that are available for each time slot and associated posted prices, he may submit a request for using the resource starting at a specific time period, according to his own preferences over starting time slots. We assume that users' token budget is adjusted to satisfy their average computing resource requirements, which are set exogenously.

The objective of the C3 manager is to *efficiently* allocate compute nodes to users. To achieve this goal, she can price the resource to shape the inter-temporal demand. Since prices are set in tokens, pricing decisions do not affect revenues directly. However, they do affect the performance of the system. Several criteria may be used to measure the system performance such as average system utilization. Close interactions with Subject Matter Experts led us to choose "fraction of satisfied customers" as the most suitable objective for our setting. We define a satisfied customer to be one who is able to obtain his preferred time slot. If a customer does obtain the time slot that is his first preference, he is deemed to be unsatisfied.

Since C3 has a pool of captive users, we assume that the manager is constrained to consider *fair* price schemes: the average price paid by a user must match the budget for the average request.

[1] We use the terms users and consumers interchangeably in the paper.

The remainder of the paper is structured as follows: Section II provides an overview of related work. In Section III, we formulate the problem. Section IV discusses a ToU pricing approach to our problem. Our simulation results are discussed in Section V.

II. Related Work

We first review the research on resource management in grid or cluster computing. We then briefly review the literature related to energy pricing that is relevant to our problem.

Buyya *et al.* (2002) compare the infrastructure required for resource allocation on a grid through different mechanisms such as exogenous prices, tenders and auctions. Vengerov (2008) proposes a reinforcement learning algorithm that tunes parameters of a sellers dynamic pricing policy even when the sellers environment is not fully observable in the context of a grid. Paleologo (2004) proposes a "Price-at-Risk" methodology that explicitly models contingent factors, such as uncertain rate of adoption or demand elasticity, to account for risk before the pricing decision is taken. To our knowledge, Yeo and Buyya (2007) are the first to consider customer utility perceived from job submission, in the context of cluster resource management. They describe an architectural framework for a utility-driven cluster resource management system. Their main contribution is the presentation a user-level job submission specification for soliciting user-centric information that is used by the cluster for making better resource allocation decisions.

Celebi and Fuller (2007) assume that suppliers in competitive electricity markets respond to prices that change hour by hour or even more frequently, but most consumers respond to price changes on a very different time scale. They examine mixed complementarity programming models of equilibrium that can bridge the speed of response gap between suppliers and consumers yet adhere to the principle of marginal cost pricing of electricity. They develop a computable equilibrium model to estimate *ex ante* Time-of-Use (ToU) prices for a retail electricity market. Borenstein *et al.* (2002) present a method for decomposing wholesale electricity payments into production costs, inframarginal competitive rents, and payments resulting from the exercise of market power. Using data from June 1998 to October 2000 in California, they find significant departures from competitive pricing, particularly during the high-demand summer months.

The literature on stock-out based substitution in retail is also related to our work. We will allude to relevant work later in the paper—see Van Ryzin and Mahajan (2001), Gaur *et al.* (2009).

III. Problem Formulation

Consider a C3 consisting of K compute nodes and N users. We will use $n \in \{1, \ldots, N\}$ to denote a generic user. We use t to index time periods that have elapsed since the beginning of the horizon. We assume that in each period t in the horizon, user n receives a positive workload $W_{n,t} := (S_{n,t}, D_{n,t})$ with probability $q_t > 0$, where $S_{n,t}$ and $D_{n,t}$ are the number of nodes required and duration in time periods. For $t \geq 1$ and $s \geq t$, let $X^t_{s,n}$ denote the number of compute nodes available at period s at the beginning of period t before user n accesses R. We assume that system workload is revealed to customers each period in an arbitrary order, and that, those that receive a positive workload access the reservation system sequentially with no preemption. Define the state of the system as $X^t_n := \{X^t_{s,n} : s \geq t\}$.

We first make a simplifying assumption on the size and duration of a request that allows us to model user preferences for starting time slots in a straightforward manner.

Assumption 3.1: The size and duration of each request is normalized to one unit, i.e., for all $t \geq 1$ and $n \in \{1, \ldots, N\}$,

$$S_{n,t} = D_{n,t} = 1.$$

Assumption 3.1 allows us to collapse the reservation system state into a (infinite dimensional) vector that indicates the number of nodes available for each future time slot. We also assume that the reservation system is prevented from making acceptance or rejection decisions, and that hard-to-model effects like fragmentation (resulting from dynamic resource allocation) are completely absent.

The remainder of this section is structured as follows: In Section III-A, we introduce the framework that we use to model customer preferences. In Section III-B, we introduce an assumption to characterize the demand that is typical in our setting. Finally, Section III-C formally introduces the objective of the C3 Manager.

A. User preferences

In order to model user preferences across different time slots, we work under the hypothesis of utility-maximizing customers. We use the Multinomial Logit choice model (MNL) framework. In particular, we assume that users receiving unit size workloads will assign (random) utilities to each upcoming time slot (including the present one) and will select the one that provides them with the highest utility among those that are still available. Let $U_n(t, s)$ denote the utility that user n

assigns to time slot s when currently in time slot t. We assume that

$$U_n(t,s) := \mu_s - p_s - \delta(s,t) + \xi^t_{n,s}, \qquad (1)$$

where p_s is the price to purchase one compute node during time slot s, μ_s is the average utility derived by a user from processing a unit workload during time slot s, $\delta(s,t)$ a discount factor depending on both time slot s and current time slot t, and $\xi^k_{n,s}$ is a (standard) Gumbel distributed random variable. We assume that these random variables are independent across t, s and n.

The modeling approach described above has been widely used in Economics, Marketing and Operations Research to model consumer substitution behavior in retail environments (for example, see Mahajan and Van Ryzin (2001)). From the user's perspective, our approach is to model different time slots as substitute products. Note that such behavior can only be justified when heterogeneity of single user preferences through time leads to an overall balance between budget and allocation: users can adopt a utility maximizing behavior as long as this does not result in average allocation spending above the predefined budget. Consequently, we assume that, if budget is allocated on a periodic basis (weekly or monthly), users are allowed to trade tokens among themselves to cover short term needs, as long as the total token balance remains non-negative.

B. Demand Structure

The overall demand is likely to exhibit patterns like the ones associated with utility services, with peak periods and off-peak periods during the day. Before we state this formally, we introduce some additional notation. Let T denote the number of periods in a day, and define $\mu := \{\mu_1, \ldots, \mu_T\}$ and $q := \{q_1, \ldots, q_T\}$ as the expected mean utility and workload arrival probability patterns, respectively, for any given day.

We next introduce some assumptions on the demand process.

Assumption 3.2: The sequence of mean utility values and workload arrivals probabilities are stationary, and consist of T different values each, i.e., $\mu_t = \mu_s$, and $q_t = q_s$ for all $t, s \geq 1$ such that $t \bmod T = s \mod T$, where $a \mod b = a - b\lfloor a/b \rfloor$. Additionally, we assume that $\delta(s,t) = \delta \cdot (\lceil t/T \rceil - \lceil s/T \rceil)$, where $\lfloor a \rfloor = \max\{b \in \mathbb{N} : b \leq a\}$, and $\lceil a \rceil = \min\{b \in \mathbb{N} : a \leq b\}$.

Assumption 3.2 will allow us to solve the pricing problem described below by solving a simpler single-day problem.

C. Decision Variables and Objective Function

The C3 manager is interested in determining the optimal pricing strategy that maximizes user satisfaction. We define a price policy to be a mapping $\pi : \mathbb{R}^\infty \to \mathbb{R}^\infty$ that specifies a sequence of prices $p_t(X,n) := \{p_t(s,X,n) : s \geq t\}$ for every system state X, where $p_t(s,X,n)$ is the price for using time slot s requested in period t when the state of the system is X and n users have submitted requests during time slot t. Let $\mathcal{P}$ denote the set of non-anticipative price policies, i.e., price policies π such that $p_t \in \mathcal{F}_t$, were $\mathcal{F}_t$ corresponds to the "history" observed up to time period t.

Recall the characterization of user satisfaction that we introduced in section I: proportion of users that get their most preferred starting slot choice. We now state this formally. Let $r(t,n) := \text{argmax}\left\{U_n(t,s) \,:\, s \geq t\,,\, X^t_{s,n} > 0\right\}$ represent the allocated time slot to user n for workload originated in time slot t if any, $r^*(t,n) := \text{argmax}\left\{U_n(t,s) \,:\, s \geq t\right\}$ represent the most preferred time slot to user n for workload that originated in time slot t, and $A^t_n := \{r^*(t,n) = r(t,n)\}$. We are interested in maximizing the steady state number of users' requests assigned to the most preferred time slot, within any T consecutive time slots, considering that the average price charged to process a request is 1, with prices measured in terms of the fraction of the budget allocated to processing a single unit workload:

$$\max_{\pi \in \mathcal{P}} \left\{ \lim_{s \to \infty} \sum_{t=1}^{T} \sum_{n=1}^{N} \mathbb{P}\left\{A^{t+s}_n \,,\, D_{n,s+t} > 0\right\} \right\} \qquad (2)$$

subject to the constraint that

$$\lim_{s \to \infty} \sum_{t=1}^{T} \mathbb{E}\left[p_t(r(s+t,n), X^t_n)\mathbf{1}\left\{D_{t,n} > 0\right\}\right] \leq N \sum_{t=1}^{T} q_t.$$

The problem stated above is intractable, even if we restrict our search to policies for which the limits above are well defined. In the next section, we will restrict our attention to a ToU class of policies.

IV. Time of Use Pricing

ToU pricing refers to the practice of charging state-independent prices that differ within a day according to the *time-of-the-day* they refer to. In what follows we restrict policies to have the ToU structure:

$$p_t(s, X^t) = f(s \mod T) \quad \forall t \text{ and } X^t.$$

Even if we restrict out attention to ToU pricing policies, (2) remains intractable due to the *stock-out based substitution phenomenon* arising from the sequential arrival of users and the limited resource capacity. Research on assortment planning addressing stock-out based substitution considers either simplified choice models that allow us to compute near-optimal solutions to the replenishment decision problem or heuristic solutions based on approximations that restore tractability. We will follow the latter path. Specifically, we will assume that each generated workload is allocated to available time slots in fixed proportions consistent with the choice model (this is the path followed by Gaur et al (2009)). To state the above formally, we need to reformulate the demand process.

A. Approximate Problem Formulation

In this section, we consider the ToU class of policies and formulate an approximation to (2) by making assumptions on the demand process. We assume that the workload is generated exclusively at the beginning of each day and can only be allocated during that day. Under this assumption, the demand and allocation across days is independent. We next make two assumptions, one on the distribution of the total workload generated during the day and the second on the allocation of the demand across different time slots during the day. These assumptions will allow us to develop a tractable formulation of (2).

Assumption 4.1: Binomial Approximation The total workload generated during a day, D, follows a Binomial distribution with parameters N and $\sum_{t=1}^{T} q_t < 1$.[2]
The reader may note that Assumption 4.1 is accurate when $\sum_{t=1}^{T} q_t << 1$.

Assumption 4.2: Constant Proportions The demand generated within a day is allocated proportionally to each available slot according to the selection probabilities derived from the MNL choice model, i.e., the demand allocated to time slot t (as a function of x, the total workload arrived) is given by

$$d_t(x) := \int_0^x \frac{\exp(\mu_t - p_t)\mathbf{1}\{d_t(y) < C\}}{\sum_{s=1}^{T} \exp(\mu_s - p_s)\mathbf{1}\{d_s(y) < C\}} dy,$$

for $x \leq D$ and $\forall\, s = 1, \ldots, T$.
Note that Assumption 4.2 essentially suppress uncertainty over users' preferences.

Let $p = (p_1, \ldots, p_T)$ denote the vector of prices during the day. For a given p, consider the permutation $\{s(1), \ldots, s(T)\}$ of $\{1, \ldots, T\}$ such that

$$\mu_{s(1)} - p_{s(1)} \geq \mu_{s(2)} - p_{s(2)} \geq \ldots \geq \mu_{s(T)} - p_{s(T)}.$$

Then, the minimum daily demand that exhausts the resource for time slot $s(t)$, $t = 1, \ldots, T$ may be written as

$$x_t = \frac{\sum_{u=t}^{T} v_{s(u)}}{v_{s(t)}} C + (t-1)C,$$

where $v_t = \exp(\mu_t - p_t)$ and $x_0 = 0$.

Lemma 4.1: Suppose assumptions 4.1 and 4.2 hold. Then, the formulation (2) reduces to

$$\min_p \left\{ \mathbb{E}\left[\sum_{t=1}^{T} (D - x_t)^+ \frac{v_{s(t)}}{\sum_{u=1}^{T} v_{s(u)}} \right] \right\}, \qquad (3)$$

subject to the constraint that

$$\mathbb{E}\left[\sum_{t=1}^{T} ((x_t \wedge D) - (x_{t-1} \wedge D)) \frac{\sum_{u=t}^{T} p_{s(u)} v_{s(u)}}{\sum_{u=t}^{T} v_{s(u)}} \right] \leq \mathbb{E}\left[D \wedge CT\right],$$

where $a^+ := \max\{a, 0\}$ and $(a \wedge b) := \min\{a, b\}$.

B. Optimal Solution for Approximate Problem

We now outline our approach to the optimization problem (3). We first introduce some additional notation. Let $\mathcal{O}$ denote the set of optimal solutions to (3), and order time slots in such a way that $\mu_1 \geq \mu_2 \geq \ldots \geq \mu_T$. Let e^t be the T dimensional vector with entries 1 in coordinates 1 through t and 0 otherwise, $t = 1, \ldots, T$. Our solution, detailed in Algorithm 1, starts from an initial feasible solution $p = 0$ and takes a sequence of steps, in each of which the objective function increases, until no further improvement is feasible. This is done by gradually increasing prices starting from the most attractive time slot and moving along until reaching the least attractive one. In each step, the algorithm looks to equalize net demand intensity of most preferred time slots while maintaining less attractive time slots with null prices, maintaining feasibility of the solution. We now discuss the intuition behind our approach. Ideally, one would like to have x_t as low as possible for all t. This is achieved by prices such that $\mu_t - p_t = k$, a constant, for all t. Under this condition, $x_t = TC \;\; \forall\, t$. Such a solution is feasible only if

$$\sum_{s=1}^{T} (\mu_s - \mu_T) \mathbb{E}\left[D \wedge CT\right] \leq \mathbb{E}\left[D \wedge CT\right]. \qquad (4)$$

The condition above holds when either demand is *low* relative to total resource capacity or when users are *highly* price-sensitive, i.e., differences in average utilities are low. Under condition (4), all components of $\mu - p$ are identical and the average price per request is less than the average budget per request. The next proposition establishes optimality of the proposed solution. The proof is omitted due to space constraints. We need some

[2]We assume that $\sum_{t=1}^{T} q_t < 1$, i.e., a user receives, on average, less than one unit of workload each day.

additional notation before we can state the result: For $t = 1, \ldots, T$, $\Delta \in \mathbb{R}^+$ and $x \in \mathbb{R}^+$ define

$$f(t, \Delta, x) := \frac{t \exp(\mu_t - \Delta) + \sum_{t+1}^{T} \exp(\mu_s)}{t \exp(\mu_t - \Delta)} C \wedge x$$

and $\delta_\mu(t) = t(\mu_t - \mu_{t-1})$, for $t = 1, \ldots, T-1$.

Algorithm 1 Optimal Solution

Set $p_t = 0$ for $t = 1, \ldots, T$.
for t =1 to $T-1$ **do**
  Define $\widehat{p}(\Delta) := p + \Delta e^t$, and set

$$\Delta_t := \quad \min \left\{ \Delta \, : \, \mathbb{E}\left[\left(\sum_{s=1}^{t-1} \delta_\mu(s) + t\Delta \right) f(t, \Delta, x) \right] \le \mathbb{E}\left[D \wedge CT \right] \right\}$$

  if $\Delta_t \geq \mu_t - \mu_{t+1}$ **then**
    $p \leftarrow \widehat{p}(\mu_t - \mu_{t+1})$
  else
    $p \leftarrow \widehat{p}(\Delta_t)$ and exit **for** loop.
  end if
end for

Proposition 4.1: The solution provided by Algorithm 1 is optimal for the approximate problem defined in (3).

V. SIMULATION

A. Preliminary Evaluation

In this section, we provide sample results of our numerical experiments to study the effectiveness of the ToU pricing policy that we propose. We evaluate and benchmark the performance of our policy on two metrics:

1) Customer Satisfaction: defined as the fraction of customers who receive their first preference request.
2) Mean Fill rate: defined as the fraction of used capacity.

We simulate a reservation system managing a 1-rack supercomputer (with 2^{10} compute nodes). The minimum demand unit is 2^5 nodes, and requests are formulated in powers of 2 units (i.e., a demand size of 3 requests for 2^8 nodes). We divided each day into 48 time slots, each of 30 minute duration. We assume that individual contributions made by time slots to the per-node utility of a user is modeled as in (1), using $\delta = 0.05$, and that the utility of an allocation is the sum of such contributions. We performed experiments with different distributions on the number of requests per time slot, the size of each request and the duration of each request.

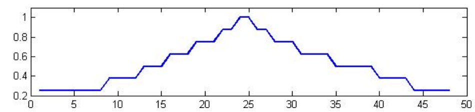

Fig. 1. Mean Utilities used for the base case (Figure 2).

Our simulations include extension and cancelation of requests. The cancelation decision is assumed to depend exclusively on the size of the request and to occur uniformly over allocation and execution times. Extension is assumed to depend exclusively on the duration of a request, while the size of the extension is uniform between one and half the original duration.

Mean utilities considered in the base case are depicted in Figure 1. We consider a base budget of 0.1 tokens per time slot and demand unit. Note that with this, Condition (4) does not hold; hence users are *highly* price sensitive. Since our pricing algorithm is based on a rather stylized model, we compute prices using an adjusted budget, such that the simulated mean price paid equals the base budget.

We provide summary statistics (average improvement in Customer Satisfaction and Mean Fill Rate) for selected problem instances. The distribution over duration and size of requests, and probabilities of requesting an extension or canceling a request, are provided in the table below. The results of the sample numerical study are summarized in Figures 2-5 below and are based on a simulation over 200 days. We limit the results that we present to the following problem instances: In our base case, the number of requests per time slot is binomially distributed with parameters $(80, 0.05)$. The results for this base case are depicted in Figure 2. We explore the effect of increased demand variance in Figure 3. We also explore the effect of decreased expected demand in Figure 4. Figure 5 depicts the effect of considering a budget of 0.5, for which equation (4) holds true.

Duration	1	2	3	4	5	6	7	8	9
Probability	0.26	0.22	0.15	0.07	0.07	0.07	0.07	0.04	0.04
Extension Prob.	0.10	0.05	0.03	0.03	0.02	0.02	0.01	0.01	0.01

Size (demand units)	0	1	2	3
Probability	0.53	0.27	0.13	0.07
Cancel Prob.	0.10	0.05	0.03	0.03

TABLE I
DISTRIBUTION OVER DURATION (IN TIME SLOTS) AND SIZE (IN $\log_2$ NODES) OF REQUESTS

We first note that the average improvement in Customer Satisfaction over the set of problem instances that we considered is 3-8 %. We next note that our pricing policy does indeed *flatten demand* not only improving user satisfaction but also enhancing mean fill rates. Further,

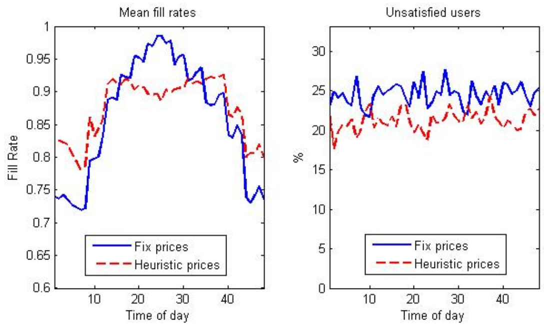

Fig. 2. Mean Fill Rate and fraction of Unsatisfied Customers for Heuristic and Fixed prices, when demand distribution is binomial $(80, 0.05)$. Results are based on simulation over 200 days.

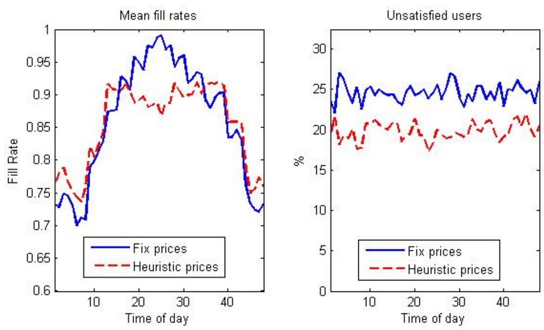

Fig. 3. Mean Fill Rate and fraction of Unsatisfied Customers for Heuristic and Fixed prices, when demand distribution is binomial $(400, 0.01)$. Results are based on simulation over 200 days.

each of three perturbations in the problem parameters (Figures 3-5) result in an improvement in Customer Satisfaction over the base case, as expected.

B. Deployment Plan

The pricing approach outlined in this paper is to be deployed as part of the Deep Cloud pilot at KAUST. This university hosts a 16-rack IBM Blue Gene/P supercomputer equipped with 4 gigabyte memory per node and capable of 222 teraflops, the fastest supercomputer in the Middle East and the *14th* on the TOP500 list worldwide. Access to the supercomputer is to be shared by a large number of faculty, students and researchers. In consultation with Subject Matter Experts, we have developed business rules to accommodate new users, cancelation and extension requests, etc. Further, a deployment plan has been developed to learn user preferences and demand.

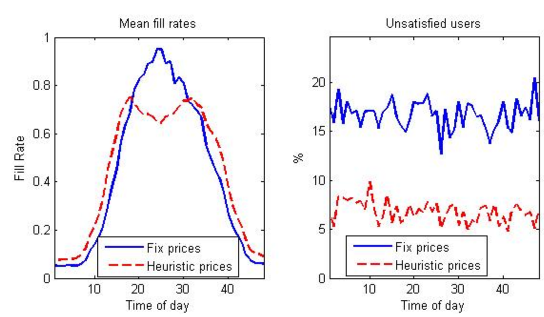

Fig. 4. Mean Fill Rate and fraction of Unsatisfied Customers for Heuristic and Fixed prices, when demand distribution is binomial $(40, 0.05)$ and budget is 0.1. Results are based on simulation over 200 days.

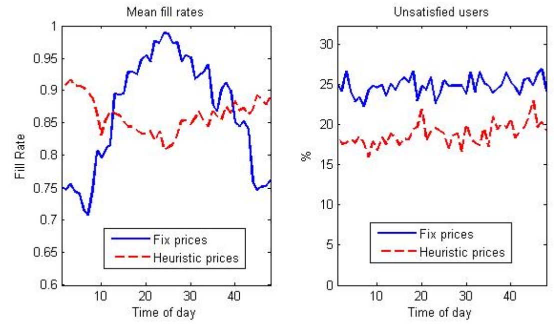

Fig. 5. Mean Fill Rate and fraction of Unsatisfied Customers for Heuristic and Fixed prices, when demand distribution is binomial $(80, 0.05)$ and budget is 0.5. Results are based simulation over 200 days.

References

[1] R. Buyya, D. Abramson, J. Giddy, and H. Stockinger. Economics Paradigm for Resource Management and Scheduling in Grid Computing. Concurrency and Computation: Practice and Experience Vol. 14, Grid Computing environments Special Issue 13-14, 2002.

[2] D. Vengerov. A gradient-based reinforcement learning approach to dynamic pricing in partially-observable environments. Future Gener. Comput. Syst. 24, 7, 687-693, 2008.

[3] G. Paleologo. 2004. Price-at-Risk: A methodology for pricing utility computing services. IBM Syst. J. 43, 1, 20-31. 2004.

[4] C. Shin Yeo and R, Buyya. 2007. Pricing for Utility-Driven Resource Management and Allocation in Clusters. Int. J. High Perform. Comput. Appl. 21, 4 (Nov. 2007), 405-418.

[5] E. Celebi, J. Fuller. A Model for Efficient Consumer Pricing Schemes in Electricity Markets. IEEE Transactions on Power Systems. Vol. 22, Issue 1 60-67. 2007.

[6] S. Borenstein, J. Bushnell, F. Wolak. Measuring Market Inefficiencies in Californias Restructured Wholesale Electricity Market. Center for the Study of Energy Markets, University of California Berkeley. 2002.

[7] S. Mahajan and G. van Ryzin. Stocking Retail Assortments Under Dynamic Consumer Substitution. Operations Research. Vol. 49, No. 3, May-June, pp. 334-351. 2001.

[8] Gartner Says Worldwide IT Spending On Pace to Surpass USD3.4 Trillion in 2008. 2008

[9] D. Honhon, V. Gaur and S. Seshadri. Assortment Planning and Inventory Decisions under stock-out based substitution. Working Paper, University of Texas at Austin. 2010.