Dynamic Cloud Pricing for Revenue Maximization

Hong Xu, *Member*, *IEEE*, and Baochun Li, *Senior Member*, *IEEE*

Abstract—In cloud computing, a provider leases its computing resources in the form of virtual machines to users, and a price is charged for the period they are used. Though static pricing is the dominant pricing strategy in today's market, intuitively price ought to be dynamically updated to improve revenue. The fundamental challenge is to design an optimal dynamic pricing policy, with the presence of stochastic demand and perishable resources, so that the expected long-term revenue is maximized. In this paper, we make three contributions in addressing this question. First, we conduct an empirical study of the spot price history of Amazon, and find that surprisingly, the spot price is unlikely to be set according to market demand. This has important implications on understanding the current market, and motivates us to develop and analyze market-driven dynamic pricing mechanisms. Second, we adopt a revenue management framework from economics, and formulate the revenue maximization problem with dynamic pricing as a stochastic dynamic program. We characterize its optimality conditions, and prove important structural results. Finally, we extend to consider a nonhomogeneous demand model.

Index Terms—Dynamic pricing, revenue maximization, spot market, cloud computing, public cloud, dynamic programming

1 Introduction

The emergence of cloud computing can already be felt with the burgeoning of cloud service offerings. Beyond technological advances, cloud computing also shows promises to the economic landscape of computing. Pricing is a crucial component of the cloud economy because it directly affects a provider's revenue and a customer's budget.

Though static pricing is the dominant strategy today, dynamic pricing emerges as an attractive alternative to better cope with unpredictable customer demand. The motivation is intuitive and simple: pricing should be leveraged strategically to influence demand to better utilize unused capacity, and generate more revenue. Indeed, Amazon EC2 [2] has introduced a "spot pricing" feature, where the spot price for a virtual instance is dynamically updated to match supply and demand as claimed in [4].

Given the flexibility to change the price on the spot, the fundamental question is, what is the *optimal* dynamic pricing policy for a provider, in terms of maximizing the expected revenue amid random demand? A provider naturally wishes to set a higher price to get a higher profit margin; yet in doing so, it also bears the risk of discouraging demand in the future. An important observation is that computing resources, such as CPU cycles and bandwidth, are inherently *perishable*: if at some point in time they are not utilized they are of no value. It is nontrivial to balance this intrinsic tradeoff with perishable capacity and stochastic demand.

- *H. Xu is with Department of Computer Science, City University of Hong Kong, Kowloon, HKSAR, China. E-mail: henry.xu@cityu.edu.hk.*
- *B. Li is with the Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Road, Toronto, ON M5S 3G4, Canada. E-mail: bli@eecg.toronto.edu.*

Manuscript received 1 May 2013; revised 14 Sept. 2013; accepted 7 Nov. 2013; published online 14 Nov. 2013.
Recommended for acceptance by A. Liang.
For information on obtaining reprints of this article, please send e-mail to: tcc@computer.org, and reference IEEECS Log Number TCC-2013-05-0083.
Digital Object Identifier no. 10.1109/TCC.2013.15.

To address this fundamental challenge, we adopt a *revenue management* framework from economics that deals with the problem of selling perishable resources, such as airline seats and hotel reservations, to maximize the expected revenue from a population of price sensitive customers [43]. Dynamic pricing has become an active field of the revenue management literature, with successful real-world applications in industries such as travel, fashion, and so on [9], [16], [38].

Cloud computing poses new challenges to solving revenue maximization problems. First, little is known about how the spot price is adjusted, and what factors are considered in the pricing algorithm, by a real-world provider such as Amazon. Also, little is known about demand statistics, and how demand reacts to price changes. In fact, though Amazon publishes its spot price history, very few insights are gained on important aspects related to modeling of the market.

Second, for a cloud provider, revenue not only depends on the number of customers, but also on the duration of usage. Unlike hotel and car rental reservations where usage durations are known, the exact usage duration of an instance in a cloud is not specified a priori. Thus, not only the arrival but also the *departure* of demand is stochastic, and has to be taken into account when collecting revenue. This clearly adds to the modeling complexity.

Our original contributions in this paper are threefold. First, we conduct an empirical study on Amazon's spot price history. We collect the spot price trace from both official and unofficial sources, spanning the time period from November 30, 2009, the inception of spot instances, to October 27, 2011, across all the regions and instance types. Surprisingly, we find that, in contrast to the common belief [13], [45], Amazon's spot price is unlikely to be set according to market supply and demand. Rather, price *oscillates* within a very narrow band most of the time, which

 Published by the IEEE CS, ComSoc, PES, CES, & SEN

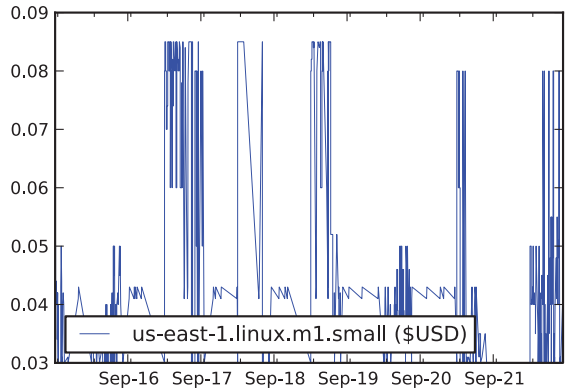


Fig. 1. Spot price of US-east-1.linux.m1.small, from 00:00:00, September 15, 2011 to 23:59:59, September 21, 2011.

is more likely to be an artifact of some pricing algorithm with predetermined reserve price. Price statistics reveal little about market demand and its relationship with price. This suggests that it is questionable to model the market based on the price statistics of Amazon.

Motivated by this observation, we consider the scenario where the cloud provider with fixed capacity updates the spot price according to market demand in this paper. Our second contribution is that we formulate the revenue maximization problem as a finite-horizon stochastic dynamic program, with stochastic demand arrivals and departures. We characterize optimality conditions for the stochastic problem and prove important structural results. Our results show that the optimal pricing policy exhibits time and utilization monotonicity, and the optimal revenue has a concave structure. They provide insights on understanding the fundamental tradeoff between pricing to the future to attract more revenue from future demand, and pricing to the present to extract more revenue from existing customers.

We also extend our model to the case with nonhomogeneous demand. We conduct an asymptotic analysis on this more general but difficult problem. We prove a surprising result that when the demand arrival and departure rates are linear with system utilization, i.e., number of existing instances, the optimal price is only a function of time and is *independent* of the system utilization.

The remainder of the paper is structured as follows: Section 2 introduces our model after an empirical study of Amazon's spot price history. In Section 3, we present our formulation and analysis of the stochastic revenue maximization problem with a homogeneous demand model. We extend the setting to a nonhomogeneous demand model in Section 4. Numerical results are provided in Section 5. In Section 6, we discuss issues pertaining to the practically of dynamic pricing, and in Section 7 we summarize related work. Finally, we conclude the paper in Section 8.

2 Model

2.1 Amazon EC2 Spot Price History

Before we introduce our model and assumptions, we wish to first do a reality check and examine the spot price history of Amazon EC2, currently the only provider that offers

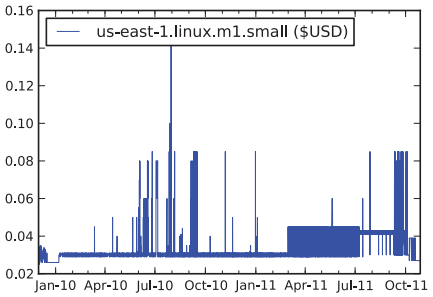


Fig. 2. Spot price of US-East-1 linux m1.small, from November 30, 2009 to October 27, 2011.

services with dynamically changing prices to our knowledge [4]. Our purpose is to obtain information on how and when a real-world provider adjusts its prices, and extract important modeling assumptions and/or constraints that we need to consider for our analysis to be meaningful.

Amazon sells virtual machines as *instances*, where different types of instances are allocated different amounts of resources. Both Windows and Linux are available. Amazon EC2 operates in seven geographical regions with different pricing [2]. Within each region, there are several *availability zones* with independent infrastructures.

Amazon starts to offer spot instances and publish spot price data in December 2009. We collect spot price traces using the ec2-describe-spot-price-history API call provided by the EC2 API tools [3]. Note that this method provides us 90 days worth of traces because Amazon only makes the most recent 90 days of spot price history publicly accessible [4]. To obtain data beyond this limitation, we download the price data accumulated by interested parties because the inception of spot instances [27], [41]. This unofficial trace runs from as early as November 30, 2009.[1] Notice that this unofficial trace is also collected using the same EC2 API method, and shall be treated with the same credibility.[2] Our combined trace has data until October 27, 2011.

While we have studied price data across all regions, availability zones, instance types, and operating systems, the results do not vary across these dimensions. To keep the presentation concise, we only show results for the Linux small standard (API name: m1.small), large standard (m1.large), as well as the extra-large high-memory (m2.xlarge) instance, in the US East (Virginia) (US-East-1) and US West (N. California) (US-West-1) regions.

We first consider a relatively short time period. Fig. 1 shows a seven-day price history for US-East-1 linux m1.small instances. Price fluctuates between $0.03 and $0.086 frequently, demonstrating that Amazon does update the price dynamically along the time line.

However, if we take a longer perspective, the story becomes drastically different. Fig. 2 shows the complete spot price history for the same linux m1.small instances in US-East-1 region. We observe temporal spikes from time to

1. Traces for certain regions such as Asia Pacific Southeast (ap-southeast) are only available from when spot instances are made available there.

2. Spot prices also vary across availability zones. However, before the 2011-05-15 version of EC2 API tools, only the lowest spot price across the region is returned from the ec2-describe-spot-price-history method. We, thus, only plot the lowest regional spot price throughout the study.

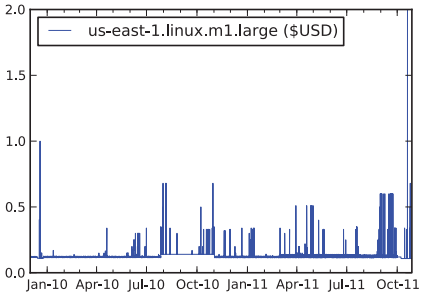


Fig. 3. Spot price of US-East-1 linux m1.large, from November 30, 2009 to October 27, 2011.

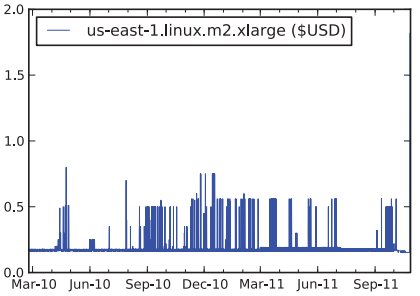


Fig. 4. Spot price of US-East-1 linux m2.xlarge, from February 23, 2010 to October 27, 2011.

time, which we conjecture correspond to the periods when capacity has to be reclaimed from spot instances to support other business. Nevertheless, it is evident the price *oscillates* within a very narrow band ($0.029-$0.031) most of the time, prior to April 2011. After that price oscillates within a wider band, again with clearly observable lower and upper bounds. Also notice that the temporal price spikes exhibit a shape of impulse bands, suggesting that price is instantly adjusted up, oscillates between a certain range, and is instantly adjusted down to the normal level.

Figs. 3 and 4 show the price history for the linux m1.large and m2.xlarge instances in the US-East-1 region. We observe the same trend where price is almost a straight line most of the time. When price spikes, it still oscillates within clearly observable upper and lower bound. Notice that in all three figures, we can easily identify a lower bound below which the price *never* goes down.

These observations imply that the Amazon spot instance market may not be a spot market where price is set according to supply and demand, as Amazon suggests [4] and many believe in their studies [13], [45]. According to Amazon, users have to submit *bids* indicating the maximum price they are willing to pay per instance per hour, and Amazon sets the price according to supply and user bids. All requests with bids higher than or equal to the spot price will run.[3] If this were the case, it is highly unlikely that the price will constantly stay bounded.[4]

Our hypothesis is that the spot price is likely to be artificially controlled with a maximum and minimum price beyond which it should never go. These maximum and minimum prices may be adjusted by Amazon according to an unknown algorithm. In fact, Ben-Yehuda et al. [8] reach the same conclusion with more thorough examination and modeling, where other possible explanations such as collaborative bidding are ruled out.

Therefore, we believe it is questionable to study dynamic pricing for a spot cloud market-based closely on the model and mechanism of Amazon, or constraints derived from its spot price data (e.g., closely bounded price). As an early theoretical work on this topic, we intentionally choose not to model the specifics of Amazon, but instead to seek to develop and analyze market-driven dynamic pricing mechanisms, which may provide insights on establishing an efficient cloud spot market in the future. As such, it is also expected that our results are different from properties of Amazon's spot price.

3. The waiting time for the spot instances to start is not guaranteed though.

4. Consider a real-world spot market, such as a commodity market for oil, cotton, and so on, or a stock exchange, where price is indeed determined by bids from buyers and sellers. The spot price evolves continuously and does not have any identifiable bounds, the existence of which would suggest arbitrage opportunities and an inefficient market [36].

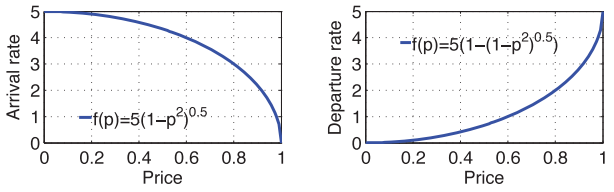
2.2 Assumptions

We now turn to introducing our model and assumptions for the spot market. We focus on an infrastructure cloud provider that sells virtual machines as instances. Its underutilized capacity is sold through the spot market with a price dynamically changing over time. Instead of using bids to determine the price, in our model the spot price is determined by the provider according to *instantaneous* demand and supply. This is simple to implement and maintain, and avoids the possible collaborative bidding to game the market. We emphasize that it is necessary for the provider to have the price-setting power, since when it needs to reclaim the capacity it can do so by raising the price and forcing out customers with low reservation prices.

We assume that the spot price p can take any value from an interval $[0, p^{\max}]$. The existence of a maximum price can be justified by the common tiered service strategy in practice. For example, Amazon offers another two tiers of products (reserved and on-demand instances) with higher QoS priorities, and the price of spot instances obviously cannot be higher than those of the higher-tier products. Price is charged per instance per time unit. Without loss of generality, we let $p^{\max} = 1$ throughout the paper. We consider a finite horizon with continuous time.

In this paper, we focus on a monopoly setting and do not consider the effect of market competition on pricing. We also assume that customers are price takers and do not consider price anticipating behavior. As the spot market for cloud computing is still at its early stage, these assumptions hold in general. Analyses of market competition and price anticipating behavior are beyond the scope of this work, and left for our future work.

The operator can influence demand by varying its price p. Demand is determined by two independent stochastic processes, namely the arrival process that corresponds to the births of new instances, and the departure process that models the deaths of existing instances (when customers shut them down). Here, we assume that demand arrivals can be expressed as a Poisson process with rate $f(p)$ (number of new instances requested per unit time). Intuitively, as price p



(a) $f(p) = k(1-p^a)^b$, where $k = 5, a = 2, b = 0.5$. (b) $g(p) = k - k(1-p^a)^b$, where $k = 5, a = 2, b = 0.5$.

Fig. 5. Examples of demand arrival and departure rate functions.

increases, customers have less financial incentives to use the service, therefore a lower arrival rate.

The demand departure process is also modeled as a Poisson process with rate $g(p)$, where $g(\cdot)$ is the departure rate function. When p decreases, customers naturally have a lower probability to leave the system, resulting in a lower departure rate $g(p)$. $f(\cdot)$ and $g(\cdot)$ can be estimated from past demand data, and this learning process can be refined periodically. This justifies our use of a finite decision horizon.

We impose several mild and practical assumptions on $f(\cdot)$ and $g(\cdot)$, which are defined over the interval $[0, 1]$. For the arrival rate function $f(\cdot)$, we assume the following properties hold:

Assumption 1. $f(p) \geq 0, f' < 0, f'' < 0, \forall 0 \leq p \leq 1$, $f'(0) = 0$, $f'(1) = -\infty$, *and* $f(1) = 0$.

The concavity of $f(\cdot)$ is a natural assumption. It reflects the common psychology that when the spot price is high, lowering it will be more attractive to customers, compared to when the spot price is already low. Other assumptions are all intuitive and reasonable requirements commonly used in the literature [9], [14], [15]. An example of such functions is $f(p) = k(1-p^a)^b$, where $k > 0, a > 1$ and $0 < b < 1$ as shown in Fig. 5a.

The departure rate function $g(\cdot)$ is clearly increasing in p. We further assume the following properties.

Assumption 2. $g(p) \geq 0, g' > 0, g'' > 0, \forall 0 \leq p \leq 1, g'(0) = \infty$, $g'(1) = 0$, *and* $g(0) = 0$.

The convexity of $g(\cdot)$ models the phenomenon that when price is high, increasing it further will have a more detrimental effect than when the price is low. An example of such functions is $g(p) = k(1-(1-p^a)^b), k > 0, a > 1$, $0 < b < 1$, as shown in Fig. 5b.

Note that the property $f(1) = 0$ allows us to model the out-of-capacity condition as an implicit constraint that forces the provider to set the price to 1 to shut down the arrival process. Similarly, $g(0) = 0$ allows the provider to price at 0 when the system is empty to turn off the departure process. They are often referred to as the "null prices" in the literature [15]. In reality, we can certainly have demand arrivals and departures without corresponding sales when the cloud is out of capacity or empty. However, in the context of our model, no generality is lost with this modeling artifact.

The Poisson assumption here is certainly an abstraction. Its use here can be justified for two reasons. First, since there is little knowledge gained on the demand model for spot instances from empirical studies, the Poisson assumption provides a good starting point for modeling with analytical tractability. Second, it is extensively used in the literature on pricing to model real-world demand processes for perishable goods, such as fashion apparel, flight seats, hotels, and so on [14], [15], [43].

3 A Stochastic Revenue Maximization Formulation

3.1 Formulation

The pricing problem can be formulated as follows: At the current time, the operator has $x \in [0, C]$ spot instances running in the system with capacity C. It faces a finite decision horizon $t > 0$ to collect revenue, until it updates the demand functions $f(\cdot)$ and $g(\cdot)$. Note here t essentially indicates *how much time is left for sale*, and decreases along the time line.

Our provider uses a nonanticipating pricing policy $p(s)$ to maximize the expected revenue over the entire decision horizon. Let $X(s)$ denote system utilization, i.e., the number of active instances in the system at any time $s \in [0, t]$. A demand is realized at time s if $\mathrm{d}X(s) = 1$, and is vanished at time s if $\mathrm{d}X(s) = -1$.

The pricing policy must be such that the number of active instances does not exceed the capacity C at any time s. We denote by $\mathcal{U}$ the set of all such possible pricing policies that satisfy

$$\int_0^s \mathrm{d}X(m) \in [-x, C-x], \tag{1}$$

$$p(s) \in [0, 1], \ \forall\, s \in [0, t]. \tag{2}$$

Here, m denotes time in $[0, s]$ when s is given. Constraint (1) is the capacity constraint mentioned above. The existence of null prices guarantees that it can always be satisfied.

Given a pricing policy $u \in \mathcal{U}$, we denote the expected revenue collected over the time period $[0, t]$ by

$$J_u(x,t) \doteq E_u\left[\int_0^t p(s)X(s)\mathrm{d}s\right], \forall t > 0. \tag{3}$$

At the very end of the horizon, when $t = 0$, the expected revenue is clearly zero for any utilization x

$$J_u(x, 0) \doteq 0, \forall x \in [0, C]. \tag{4}$$

The provider's problem is to find a pricing policy u^* that maximizes the expected revenue generated over $[0, t]$, denoted by $J^*(x, t)$. Equivalently,

$$J^*(x,t) \doteq \sup_{u \in \mathcal{U}} J_u(x,t). \tag{5}$$

3.2 Optimality Conditions

Equation (5) is a stochastic dynamic programming problem. To solve it, we can consider its Hamilton-Jacobi conditions, which are the continuous-time counterpart of the Bellman equation. Informally, consider what happens over a small interval of time δt. Since both the arrival and departure processes are Poisson, by selecting a price p, the provider sees one more instance over the next δt with

probability $f(p)\delta t + o(\delta t)$, one fewer instance with probability $g(p)\delta t + o(\delta t)$, and no change with the rest of the probability mass. By the Principle of Optimality,

$$\begin{aligned} J^*(x,t) = \sup_p \big[& px\delta t + o(\delta t) \\ & + f(p)\delta t \cdot J^*(x+1, t-\delta t) \\ & + g(p)\delta t \cdot J^*(x-1, t-\delta t) \\ & + (1 - (f(p)+g(p))\delta t) J^*(x, t-\delta t)\big]. \end{aligned} \tag{6}$$

In words, with t time left to the end of the horizon, the optimal expected revenue $J^*(x,t)$ must be equal to the realized revenue during δt, which is simply $px\delta t$, plus the expected value of the optimal expected revenue from the remaining time interval $t - \delta t$, which are the remaining terms of (6).

Rearranging the terms and taking the limit as $\delta t \to 0$, we get

$$\begin{aligned} \frac{\partial J^*(x,t)}{\partial t} = \sup_p \big[& px + f(p)(J^*(x+1,t) - J^*(x,t)) \\ & - g(p)(J^*(x,t) - J^*(x-1,t))\big]. \end{aligned} \tag{7}$$

Note that (7) holds only for $1 \le x \le C-1$. When $x = 0$, the provider will not see any departure over δt, and is forced to price at 0; when $x = C$ the provider is forced to set the price to 1 to shut down the arrival process as discussed above. Thus, $p^*(0,t) = 0$ and $p^*(C,t) = 1$ in our model. We have the following:

$$\begin{aligned} J^*(0,t) &= f(0)\delta t \cdot J^*(1, t-\delta t) \\ &\quad + (1 - f(0)\delta t) J^*(0, t-\delta t) + o(\delta t), \\ J^*(C,t) &= g(1)\delta t \cdot J^*(C-1, t-\delta t) \\ &\quad + (1 - g(1)\delta t) J^*(C, t-\delta t) + C\delta t + o(\delta t), \end{aligned}$$

from which we obtain the following conditions:

$$\frac{\partial J^*(0,t)}{\partial t} = f(0)\big(J^*(1,t) - J^*(0,t)\big), \tag{8}$$

$$\frac{\partial J^*(C,t)}{\partial t} = C - g(1)\big(J^*(C,t) - J^*(C-1,t)\big). \tag{9}$$

We have not yet justified interchanging $\sup_p$ and $\lim_{\delta t \to 0}$. This can be done formally using [10, Theorem 2.1]. This is also reminiscent to the technique used in [15]. Thus, a solution to (7) with boundary conditions (4) is indeed the optimal revenue $J^*(x,t)$, from which we can readily obtain the optimal prices $p^*(x,t)$ that together form an optimal pricing policy u^*.

We now show the existence of a unique solution to (7).

Proposition 1. *If the demand arrival and departure rate functions f and g satisfy Assumptions 1 and 2, there exists a unique solution to (7) with boundary conditions (4).*

Proof. The optimal price p is always within the compact set $[0,1]$. Combining compactness with the fact that f and g are continuous and bounded in p establishes the conditions required by [10, Theorem 2.3] for the existence of a unique solution to (7). □

3.3 Structural Results

Although we have found the optimality conditions, solving them to obtain a closed-form solution is quite difficult for arbitrary demand arrival and departure functions. Moreover, numerically computing the optimal solution can also be prohibitive as the state space grows exponentially with the capacity of the cloud provider, which is typically fairly large. However, we are able to characterize several important structural properties of the optimal solution to the dynamic program (7). We believe that insights obtained from our analysis are fundamental in understanding the problem, and instrumental toward designing computationally efficient heuristics, which is important in practice.

Theorem 1 (Monotonicity of optimal revenue). *$J^*(x,t)$ is strictly increasing in both x and t.*

Proof. The fact that $J^*(x,t)$ is strictly increasing in t is intuitive and can be straightforwardly proved, and we omit the details here. The fact that $J^*(x,t)$ is also increasing in x is not trivial because with a smaller number of running instances x to start with, the provider has an incentive to set a lower price to attract more customers, and the net effect on revenue can be either positive or negative.

By definition (3), we can write

$$\begin{aligned} & J^*(x,t) \\ & = E_u\left[\int_0^t p_u(s) x \mathrm{d}s\right] + E_u\left[\int_0^t p_u(s) \int_0^s \mathrm{d}X_u(m)\mathrm{d}s\right], \end{aligned} \tag{10}$$

where u is the optimal policy and clearly satisfies

$$\int_0^s \mathrm{d}X_u(m) \in [-x, C-x], \forall s \in [0,t].$$

$\mathrm{d}X_u(m)$ is the optimal birth-death process that corresponds to the optimal policy u. Alternatively, we can also think of $p_u(m)$ as determined by the statistics of the optimal birth-death process $\mathrm{d}X_u(m)$ at time m, through $E[\mathrm{d}X_u(m)/\mathrm{d}m] = f(p_u(m)) - g(p_u(m))$.

Now, we let $\mathrm{d}X_v(m)$ be another birth-death process that relates to $X_u(m)$ by

$$\int_0^s \mathrm{d}X_v(m) = \int_0^s \mathrm{d}X_u(m) - 1, \forall s \in [0,t].$$

$\mathrm{d}X_v(m)$ corresponds to another pricing policy, denoted by v, which can be obtained from the following relationship:

$$f(p_v(m)) - g(p_v(m)) = E[\mathrm{d}X_v(m)/\mathrm{d}m].$$

Obviously, $\mathrm{d}X_v(m) \le \mathrm{d}X_u(m)$ holds at all times $m \in (0,s]$. Further, $\mathrm{d}X_v(m) < \mathrm{d}X_u(m)$ must hold for at least $m = 0$. Thus,

$$\begin{aligned} & E[\mathrm{d}X_v(m)] \le E[\mathrm{d}X_u(m)] \Rightarrow p_v(m) \ge p_u(m), \forall m \in (0,s], \\ & E[\mathrm{d}X_v(0)] < E[\mathrm{d}X_v(m)] \Rightarrow p_v(0) > p_u(0). \end{aligned} \tag{11}$$

We can write out the expected revenue starting with $x+1$ instances under the policy v as follows:

$$
\begin{aligned}
&J_v(x+1,t)\\
&= E_v\left[\int_0^t p_v(s)(x+1)\mathrm{d}s\right] + E_v\left[\int_0^t p_v(s)\int_0^s \mathrm{d}X_v(m)\mathrm{d}s\right]\\
&= E_v\left[\int_0^t p_v(s)x\mathrm{d}s\right] + E_v\left[\int_0^t p_v(s)\left(1+\int_0^s \mathrm{d}X_v(m)\right)\mathrm{d}s\right]\\
&= E_v\left[\int_0^t p_v(s)x\mathrm{d}s\right] + E_v\left[\int_0^t p_v(s)\int_0^s \mathrm{d}X_u(m)\mathrm{d}s\right],
\end{aligned}
\tag{12}
$$

where v is a feasible policy because

$$\int_0^s \mathrm{d}X_v(m) \in [-x-1, C-x-1], \forall s \in [0,t].$$

Comparing to (10), it is readily seen that $J_v(x+1,t) > J^*(x,t)$ due to (11). Thus, $J^*(x+1,t) > J^*(x,t)$. □

Theorem 1 asserts that the optimal expected revenue increases with the utilization of the system and/or time. Moreover, we can show that the optimal revenue exhibits diminishing marginal returns with respect to utilization.

Theorem 2 (Concavity of optimal revenue). *$J^*(x,t)$ is concave in x for any fixed $t > 0$.*

Proof. It suffices to show that

$$2J^*(x,t) \geq J^*(x+1,t) + J^*(x-1,t), \forall t > 0 \tag{13}$$

holds for all $x \in [1, C-1]$, which we prove by constructing a feasible policy to solve $J^*(x,t)$ with expected revenue equal to $(J^*(x+1,t) + J^*(x-1,t))/2$.

Suppose u and v are optimal policies that achieve $J^*(x+1,t)$ and $J^*(x-1,t)$, respectively. Denote the corresponding optimal birth-death processes as $\mathrm{d}X_u(m)$ and $\mathrm{d}X_v(m)$, respectively. Now, consider a new birth-death process as follows:

$$\mathrm{d}X_{u'}(m) = \frac{\mathrm{d}X_u(m) + \mathrm{d}X_v(m)}{2}, \forall m \in [0,t).$$

This new process corresponds to a policy u', where $p_{u'}(m) \geq (p_u(m) + p_v(m))/2$ due to the concavity of the function $E[\mathrm{d}X(m)/\mathrm{d}m] = f(p(m)) - g(p(m))$ by Assumptions 1 and 2. By construction, $\int_0^s \mathrm{d}X_{u'}(m) \in [-x, C-x]$ holds for all $s \in [0,t]$. Thus, u' is a feasible solution for $J^*(x,t)$. Readily,

$$J^*(x,t) \geq J_{u'}(x,t) \geq \frac{J^*(x+1,t) + J^*(x-1,t)}{2}.$$

□

This concavity property of the optimal revenue is not only crucial for further development of this paper but also of interest in itself. For example, it can be useful for determining the optimal number of instances running in the system if it is part of the decisions. When the cost of providing computing hardware is linear or strictly convex, the expected profit becomes a concave function of the number of running instances. In this case, the optimal utilization is the largest quantity for which the marginal expected revenue exceeds the marginal cost.

We proceed to consider how the optimal price changes over time and the utilization x. Our first result is that the optimal price increases with the system utilization.

Theorem 3 (Utilization monotonicity of optimal price). *$p^*(x,t) < p^*(x+1,t)$ for any fixed t, $x \in [0, C-1]$.*

Proof. For convenience, let us denote $M(x,t) = J^*(x,t) - J^*(x-1,t)$. From Theorem 2, we know that $M(x+1,t) \leq M(x,t)$. If we take the derivative of the right side of (7) with respect to p and set it to zero, we get the necessary and sufficient condition for $p^*(x,t)$ which we abbreviate as p_x^*:

$$x + f'(p_x^*)M(x+1,t) - g'(p_x^*)M(x,t) = 0. \tag{14}$$

By Assumptions 1 and 2, we have

$$
\begin{aligned}
&g'(p_x^*)M(x,t) - f'(p_x^*)M(x,t) \geq x\\
\Longrightarrow &M(x,t) \geq \frac{x}{g'(p_x^*) - f'(p_x^*)}\\
\Longrightarrow &M(x+1,t) \geq \frac{x+1}{g'(p_{x+1}^*) - f'(p_{x+1}^*)}.
\end{aligned}
$$

Similarly, we have

$$
\begin{aligned}
&g'(p_x^*)M(x+1,t) - f'(p_x^*)M(x+1,t) \leq x\\
\Longrightarrow &M(x+1,t) \leq \frac{x}{g'(p_x^*) - f'(p_x^*)}.
\end{aligned}
$$

Thus, for the two inequalities to hold, we must have

$$
\begin{aligned}
&\frac{x+1}{g'(p_{x+1}^*) - f'(p_{x+1}^*)} \leq \frac{x}{g'(p_x^*) - f'(p_x^*)}\\
\Longrightarrow &g'(p_{x+1}^*) - f'(p_{x+1}^*) > g'(p_x^*) - f'(p_x^*)\\
\Longrightarrow &p^*(x+1,t) > p^*(x,t).
\end{aligned}
$$

□

Theorem 3 has natural economic interpretations. When the system is heavily loaded, it is in the interest of the provider to set a higher price to obtain a higher revenue from customers, as well as to discourage future demand to prevent the system from overloading. On the other hand, when the system is lightly utilized, the provider can afford to adopt a lower price to attract more customers.

We further show that the optimal price also exhibits time monotonicity. That is, $p^*(x,t)$ is decreasing in t. To prove this result, we first need a technical lemma.

Lemma 1. *$\frac{\partial M(x,t)}{\partial t} > 0$ for any given t, where $M(x,t) = J^*(x,t) - J^*(x-1,t)$.*

Proof. Note that at $t = 0$, $J^*(x,0) = 0$ for all $x \in [0, C]$. Thus, $M(x,0) = 0$. Assume that $\frac{\partial M(x,t)}{\partial t} \leq 0$. Then, at some $t' > 0$, $M(x,t') \leq 0$, which contradicts with Theorem 1. Thus, the lemma must hold. □

Theorem 4 (Time monotonicity of optimal price). *$p^*(x,t)$ is decreasing in t for all $x \in [1, C-1]$.*

Proof. It suffices to prove that

$$\frac{\partial p^*(x,t)}{\partial t} < 0, \forall x \in [1, C-1]. \tag{15}$$

Taking derivative with respect to t in (14) and rearranging the terms, we have

$$\frac{\partial p^*(x,t)}{\partial t}(g''(p^*(x,t))M(x,t) - f''(p^*(x,t))M(x+1,t)) = f'(p^*(x,t))\frac{\partial M(x+1,t)}{\partial t} - g'(p^*(x,t))\frac{\partial M(x,t)}{\partial t}.$$

Applying Lemma 1 and Assumptions 1 and 2, the RHS is seen to be negative. Since $g'' > 0, f'' < 0, M(x,t) \geq 0$, $\frac{\partial p^*(x,t)}{\partial t}$ must be negative in the LHS. □

Therefore, as time runs out, the optimal price is increasing, and at the end of the horizon when $t = 0$, $p^*(x,0)$ is readily found to be 1, the maximum possible price, for all x since $M(x,0) = 0$ in (7). The intuition is that when the provider has a long period of time left (t is large), she should price to the future and set a lower price to attract more customers to maximize the expected revenue. As time goes, her focus is shifting to the existing customers, and the optimal strategy is to set a higher price to extract more revenue. At the end, when $t = 0$, it simply sets price to maximize the current revenue and entirely ignores the impact on future revenue.

The properties that we proved in Theorems 1, 2, 3, and 4 are not only intuitively satisfying, but also useful if we wish to compute the optimal policy numerically because they significantly reduce the search space of policies over which one needs to optimize.

4 Nonhomogeneous Demand

In the preceding analysis, we have assumed a time homogeneous demand model, where the demand arrival and departure rates are time invariant functions of price. This assumption is restrictive. In this section, we study the general case where both the demand arrival and departure functions may change over time.

To keep the analysis tractable, we adopt a simple nonhomogeneous demand model. We assume that demand arrival at time s can be expressed as a Poisson process with a rate $X(s)f(p(s))$. $X(s)f(p(s))$ denotes the price-sensitive demand arrival rate with $f(p(s))$ being the endogenous arrival probability function. We use $X(s)$ as a linear multiplier to model the intuition that when $p(s)$ decreases, each existing customer has an increased probability $f(p(s))$ to shift more workload to the cloud, resulting in more demand.

The demand departure process is similarly modeled as a Poisson process with a time-varying rate function $X(s)g(p(s))$, where $g(p(s))$ is the endogenous departure probability. When $p(s)$ decreases, each customer has a lower probability $g(p(s))$ to leave the system, resulting in a lower rate $X(s)g(p(s))$. The same structural properties as assumed in Section 3 are carried over here, namely:

Assumption 3. $0 \leq f(p) \leq 1, f' < 0, f'' < 0, \forall 0 \leq p \leq 1$, $f'(0) = 0, f'(1) = -\infty, f(0) = 1$, *and* $f(1) = 0$.

Assumption 4. $0 \leq g(p) \leq 1, g' > 0, g'' > 0, \forall 0 \leq p \leq 1$, $g'(0) = \infty, g'(1) = 0, g(0) = 0$, *and* $g(1) = 1$.

4.1 Formulation

The revenue maximization problem under such a nonhomogeneous demand model can then be formulated as follows similar to (5):

$$J^*(x,t) \doteq \sup_{u \in \mathcal{U}} \int_0^t p(s)X(s)\mathrm{d}s, \quad \text{s.t. } X(s) = x + \int_0^s \mathrm{d}X(m), \tag{16}$$

where $\mathcal{U}$ denotes the set of admissible policies that satisfy $\int_0^s \mathrm{d}X(m) \in [-x+1, C-x]$ for all $s \in [0,t]$. Notice that in this model, $X(s)$ must be larger than or equal to 1. Thus, the provider is forced to set $p^*(1,t) = 0$, and we are only interested in cases where $x \geq 2$. No generality is lost in making this modeling artifact.

Similarly, we obtain the optimality conditions as follows:

$$\frac{\partial J^*(x,t)}{\partial t} = \sup_p \; x\big[p + f(p)(J^*(x+1,t) - J^*(x,t)) - g(p)(J^*(x,t) - J^*(x-1,t))\big]. \tag{17}$$

The necessary and sufficient condition for $p^*(x,t)$ is then

$$1 + f'(p_x^*)M(x+1,t) - g'(p_x^*)M(x,t) = 0. \tag{18}$$

4.2 Asymptotic Analysis

The nonhomogeneous model creates additional difficulty in obtaining structural properties regarding the optimal pricing policy. To overcome the difficulty, while keeping the results relevant, we conduct an asymptotic analysis here. Specifically, we assume $C \to \infty$, and drop the capacity constraint that limits the set of admissible policies $\mathcal{U}$. This models a large-scale problem, where the capacity of a cloud is always enough to accommodate all the virtual instances, and thus, the capacity constraint is usually inactive. This is also useful as an approximate solution to the original problem (16), as the asymptotic optimal solution turns out to have a very simple structure.

Our first result is that the monotonicity of optimal revenue still holds in the nonhomogeneous demand model.

Theorem 5. *$J^*(x,t)$ is strictly increasing in both x and t in the nonhomogeneous demand model.*

Proof. The fact that $J^*(x,t)$ is strictly increasing in t is intuitive. We only prove for the result that $J^*(x,t)$ is also increasing in x.

Consider $J^*(x,t)$ and $J^*(x-1,t)$. Let u_2 be the optimal pricing policies that achieve $J^*(x-1,t)$. Clearly, u_2 is feasible for $J^*(x,t)$. From (3), we have

$$\frac{J_{u_2}(x,t)}{J^*(x-1,t)} = \frac{\int_0^t p(s)E_{u_2}[X_1(s)]\mathrm{d}s}{\int_0^t p(s)E_{u_2}[X_2(s)]\mathrm{d}s}.$$

$E_{u_2}[X_1(s)], E_{u_2}[X_2(s)]$ can be found to grow exponentially over time, i.e.,

$$E_{u_2}[X_1(s)] = x \cdot e^{q(s)}, \quad E_{u_2}[X_2(s)] = (x-1)e^{q(s)}, \quad \text{where } q(s) = \int_0^s \big(f(p(s)) - g(p(s))\big)\mathrm{d}s. \tag{19}$$

Thus,

$$\frac{J^*(x,t)}{J^*(x-1,t)} \geq \frac{J_{u_2}(x,t)}{J^*(x-1,t)} = \frac{x}{x-1} > 1.$$

□

Moreover, we can prove that the optimal pricing policy is *independent* of the number of instances in the system.

Theorem 6. $p^*(x,t) = p^*(t), \forall x \geq 2$.

Proof. From the proof of Theorem 5, we know that $\frac{J^*(x,t)}{J^*(x-1,t)} \geq \frac{x}{x-1}$. By the same token, assume that u_1 is the optimal pricing policy that achieves $J^*(x,t)$. Since the constraint that $\int_0^s \mathrm{d}X(m) \in [-x+1, C-x]$ is dropped, u_1 is also feasible for $J^*(x-1,t)$. Then, we have

$$\frac{J^*(x,t)}{J^*(x-1,t)} \leq \frac{J^*(x,t)}{J_{u_1}(x-1,t)} = \frac{x}{x-1}. \tag{20}$$

Thus, $\frac{J^*(x,t)}{J^*(x-1,t)} = \frac{x}{x-1}$, and u_1 must be equal to u_2 for the equation to hold, thus the proof. □

Theorem 6 is surprising. It tells us that when the system capacity is not a concern, the optimal pricing policy with nonhomogeneous demand is only a function of the time remaining to the end of the horizon. This is so because the expected demand is no longer upper bounded by the system capacity. It is always optimal to maximize the expected revenue *solely* obtained from future demand, which is completely determined by the time remaining to the end of the horizon and does not depend on the utilization. The optimal expected revenue using the same pricing policy is, thus, proportional to x as $E[X(s)]$ is proportional to x from (19). This property greatly reduces the complexity of solving the stochastic dynamic program by an order of C because time is the only state variable.

Further, we can show that for a given number of active instances, the optimal price still exhibits time monotonicity.

Theorem 7. *There exists an optimal price $p^*(t)$ that is strictly decreasing in t.*

Proof. Since $\frac{J^*(x,t)}{J^*(x+1,t)} = \frac{x}{x+1}$, $M(x,t) = M(x+1,t) = M(t)$ where $M(x,t) = J^*(x,t) - J^*(x-1,t)$. Substitute into (18),

$$M(t) = \frac{1}{g'(p^*(t)) - f'(p^*(t))}.$$

We denote $h(p) = \frac{1}{g'(p)-f'(p)}$. Taking derivative with respect to t, we have

$$M'(t) = h'(p) \cdot (p^*)'(t).$$

From the definition of $M(t)$ and (17),

$$\begin{aligned} M'(t) &= \frac{\partial J^*(x,t)}{\partial t} - \frac{\partial J^*(x-1,t)}{\partial t} \\ &= M(t)(f(p^*(t)) - g(p^*(t))) + p^*(t) = \frac{\partial J^*(x,t)}{x \partial t} > 0. \end{aligned}$$

Since $h' = \frac{r''(g'-f') - (g''-f'')r'}{(g'-f')^2}$, with Assumptions 3, 4, $h' < 0$. Thus, $\frac{\partial p^*(t)}{\partial t} < 0$. □

5 Numerical Studies

In this section, we conduct numerical studies to verify the properties of the optimal pricing policy. The system capacity C is set to 10,000. This corresponds to a moderate scale data center, such as a single availability zone of Amazon EC2, with several thousand machines capable of running tens of thousands of instances [5]. The decision horizon t is normalized to $[0,1]$, which corresponds to a 1-hour period, and prices are charged on a usage time basis as we discussed in Section 2.2.

We follow the standard approach of numerically solving a continuous time dynamic program—discretizing the time horizon into N intervals of length Δt and using a difference equation to approximate the optimality equation. The resulting difference equation

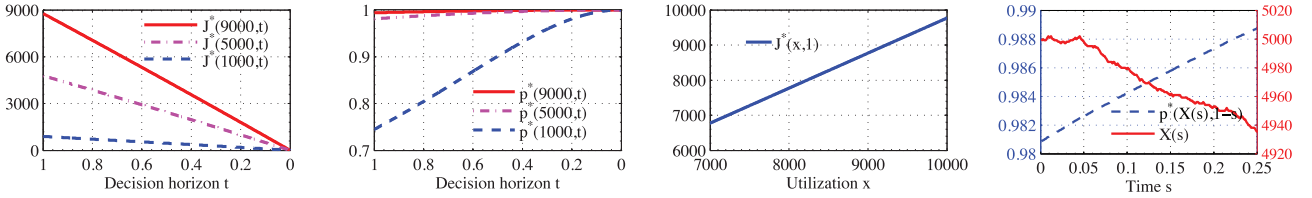
$$\begin{aligned} J^*(x,t) = \max_{p \in [0,1]} \big[& px\Delta t + f(p)\Delta t J^*(x+1, t-\Delta t) \\ & + g(p)\Delta t J^*(x-1, t-\Delta t) \\ & + (1 - f(p)\Delta t - g(p)\Delta t) J^*(x, t-\Delta t) \big] \end{aligned}$$

can be solved by backward induction for the discrete time set $\{n\Delta t \mid n = 0, 1, \ldots, N\}$ with boundary conditions $J^*(x,0) = 0$. We consider demand functions of the form $f(p) = k\sqrt{(1-p^2)}$ and $g(p) = k - k\sqrt{(1-p^2)}$ as shown in Fig. 5. In this case, the optimal price has a closed-form solution $p^*(x,t) = y/\sqrt{k^2+y^2}$, where $y = x/(J^*(x+1, t-\Delta t) - J^*(x-1, t-\Delta t))$. To be accurate, the total number of time intervals $N = 1/\Delta t$ should be much larger than the maximum number of demand arrivals and departures k. The time and space complexity of the computation are both $\mathrm{O}(CN)$.

5.1 Weak Dynamics Scenarios

We first consider a weak dynamics scenario, where the maximum expected demand arrivals and departures during the decision horizon k is orders of magnitude less than the system capacity C. We assume that a cloud is expected to launch and close several hundreds of instances per hour on average. Thus, we set k to 500, much smaller than the system capacity $C = 10{,}000$. Time is discretized into $N = 1{,}000$ intervals, each corresponding to 3.6 seconds of time for a 1-hour horizon. The results are shown in Fig. 6. The optimal expected revenue clearly grows with decision horizon t (which decreases with time in the figures) and the utilization x as seen from Fig. 6a. The optimal price decreases with t (increases with time), and increases with the utilization as seen from Fig. 6b. These observations validate our analysis in Section 3.

Note that the optimal price does not change much when x decreases from 9,000 to 5,000, and is close to 1 for the entire horizon. This is because the effect of demand dynamics is small compared to a moderately loaded system ($x = 5{,}000 = 0.5C$), and the expected revenue can be maximized without considering much about the future demand, i.e., setting price close to 1. To facilitate the understanding, Fig. 6d shows a sample path of the optimal price, with the corresponding system utilization process $X(s)$ starting from $x = 5{,}000$. In the time period of [0,0.25] (t decreases from 1 to 0.75), $p^*(X(s), 1-s)$ grows only



(a) Optimal revenue. (b) Optimal price. (c) Optimal revenue vs. utilization at $t = 1$. (d) A sample path of $X(s)$ and the corresponding optimal price $p^*(x,t)$.

Fig. 6. Numerical results with $C = 10{,}000$, $f(p) = 500\sqrt{1-p^2}$, $g(p) = 500 - 500\sqrt{1-p^2}$, and $N = 1{,}000$ time intervals.

marginally from 0.98 to around 0.988, while $X(s)$ slowly decreases from 5,000 to about 4,930.

Thus, $X(t)$ does not deviate much from x, and $J^*(x,t)$ is close to xt, which also explains its linearity as in Fig. 6c. When the system is lightly loaded ($x = 1{,}000 = 0.1C$), revenue generated from future demand becomes more important, and $p^*(x,t)$ is much lower and varies with time as in Fig. 6b.

In summary, these results tell us that when the expected dynamics is weak, or equivalently when the decision horizon is short, it is *almost* optimal to use a static price close to the maximum price 1 for a heavily or moderately loaded system. The optimal expected revenue grows linearly with the utilization. When the system is lightly loaded, however, price has to be dynamically adjusted to obtain maximum revenue.

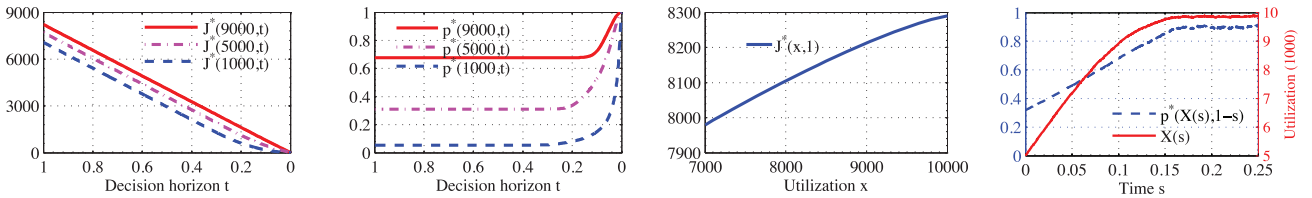
5.2 Strong Dynamics Scenarios

The story becomes quite different when the problem embraces a significant degree of demand dynamics. Here, we let the maximum expected demand arrivals and departures k equal to 50,000, much larger than the system capacity C. Time is discretized into $N = 10^5$ intervals. Other parameters remain the same. The results are shown in Fig. 7.

The optimal revenue and price clearly exhibit monotonicity as expected from our analysis. Compared to the weak dynamics case, the first interesting observation is that optimal revenue is *insensitive* to the utilization. As seen from Fig. 7a, $J^*(9{,}000,t)$ improves $J^*(1{,}000,t)$ only by a small constant margin, and xt becomes a poor estimate for $J^*(x,t)$, especially when t is close to 1. The reason is that when the demand dynamics is strong, revenue from future demand is dominant especially in the beginning of the horizon (when t is close to 1). Since price can be adjusted flexibly, the system can always be quickly tuned to a heavy load setting with better revenue, and the end result is that the expected revenue over the horizon is relatively immaterial to the current utilization. This also explains the stronger concavity of $J^*(x,t)$ in x as seen in Fig. 7c because the marginal benefit of increasing the utilization is diminishing, causing the revenue curve to be bent downwards.

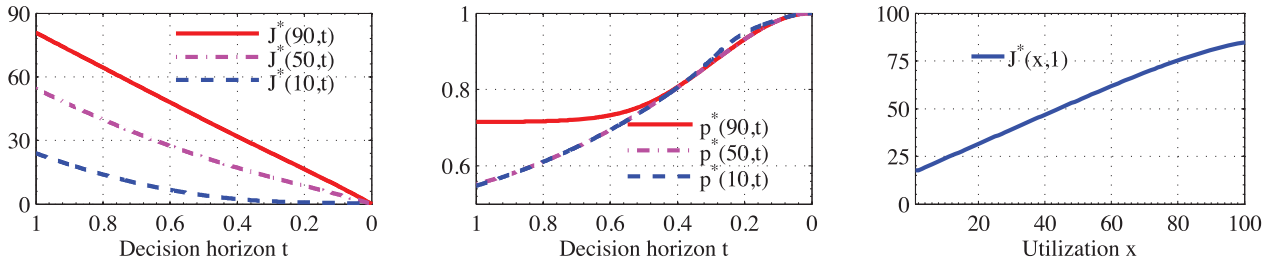
The discussion above implies that dynamic pricing becomes more critical in a strong dynamics setting. It is, thus, expected to see the optimal price varying significantly with the utilization, which is demonstrated in Fig. 7b. Compared with Fig. 6b, for most of the time $p^*(1{,}000,t)$ remains to be much smaller than $p^*(5{,}000,t)$ which in turn is much smaller than $p^*(9{,}000,t)$. The reason is that when the effect of dynamics is significant, we should price to the future even when time left to consider t is relatively small, and adopt a low price when the utilization is low. Only when it is near the end of the horizon, should we raise the price to harvest more revenue from the existing customers.

A critical point here is that although $p^*(x,t)$ remains almost static most of the time for a given x, it does not mean that we can safely use a static price at a small cost of revenue loss. In fact, the number of instances in the system is expected to fluctuate quickly over time, and whenever it changes we ought to change the price on the spot. Since $p^*(x,t)$ differs widely with x, we ought to use a dynamic pricing policy to maximize the revenue. This is demonstrated in Fig. 7d with a sample path of the optimal price. We can see that the utilization process $X(s)$ quickly grows from 5,000 to nearly 10,000, and the optimal price $p^*(X(s),1-s)$ rises from about 0.3 to over 0.8 in the time period of $[0, 0.25]$. Compared with Fig. 6d for the weak dynamics case, the optimal price is clearly much more dynamic. It is reasonable to conclude that dynamic pricing plays an important role in the strong dynamics setting,



(a) Optimal revenue. (b) Optimal price. (c) Optimal revenue vs. utilization at $t = 1$. (d) A sample path of $X(s)$ and the corresponding optimal price $p^*(x,t)$.

Fig. 7. Numerical results with $C = 10{,}000$, $f(p) = 50{,}000\sqrt{1-p^2}$, $g(p) = 50{,}000 - 50{,}000\sqrt{1-p^2}$, and $N = 10^5$ time intervals.



(a) Optimal revenue. (b) Optimal price. (c) Optimal revenue vs. utilization at $t = 1$.

Fig. 8. Numerical results with $C = 100, f(p) = \sqrt{1-p^2}, g(p) = 1 - \sqrt{1-p^2}$, and $N = 1{,}000$ time intervals.

and is expected to offer substantial revenue improvement over static pricing.

5.3 Sensitivity Analysis

In the previous two sections, we studied the cases where the maximum expected demand arrivals and departures k is set to 500 and 50,000, respectively. These represent two extreme cases, and naturally one may wonder if our observations above are sensitive to the values of k, in the sense that the results are skewed at the extreme points. In this section, we provide evidence to confirm that the observations are robust to k.

In this experiment, we vary k from 500 to 10,000, and discretize time into 10,000 intervals. Other parameters are the same as in the previous two sections. Fig. 10 shows the optimal price at 50 percent utilization $p^*(5{,}000, t)$ for different values of k. Observe that when k is small compared to C, $p^*(5{,}000, t)$ behaves qualitatively similar to the curve in Fig. 6b. As k increases, optimal price decreases. The curve gradually becomes flatter, meaning that most of the time the optimal price should be kept low until at the end of the horizon. The shape of the curve also becomes qualitatively similar to Fig. 7b for the strong dynamics case.

Thus, our results drawn from two extreme values of k are also valid for intermediate values of k. We also studied the effect of k for optimal revenue J^* and the observation is the same. For brevity, we omit the figure.

5.4 Impact of Delay

We have assumed that the provider always has perfect information about the system utilization of the cloud at any time. In reality, the infrastructure monitoring software for the cloud may incur delay in processing and propagating data, especially given the large scale of the system. Delayed information inherently limits the provider's ability to make correct pricing decisions, and causes revenue loss. In this section, we investigate the impact of delay on provider's revenue.

We conduct a set of numerical studies with $k = 10{,}000$, and $N = 1{,}000$ intervals for $T = 1$. We consider a moderately loaded system with an initial utilization of $x = 5{,}000$, with varying information delay δ ranging from 0.001 to 0.01. We consider the time period of $[0, 0.25]$. At each time interval s, the provider sees a delayed utilization $X(s-\delta)$ instead of $X(s)$, and makes a pricing decision based on $X(s-\delta)$. For each value of delay, we generate 50 sample paths of the system utilization process $X(s)$ and pricing decisions $p^*(X(s-\delta), 1-s)$, and compute the average revenue for the period $[0, 0.25]$ across the 50 runs.

Fig. 11 plots the percentage of revenue loss due to delay. As expected, revenue loss increases when delay is more salient, since with a long delay the actual system utilization is significantly different from what the provider sees. We observe that when delay is small, i.e., less than 0.004, revenue loss is less than or around 1 percent. However, when delay is larger than 0.004, revenue loss is larger than 2 percent. In reality information, delay is usually small compared to the time horizon T. For example, T is 1 hour in our numerical study, and a 1-second delay is only $0.00026T$.

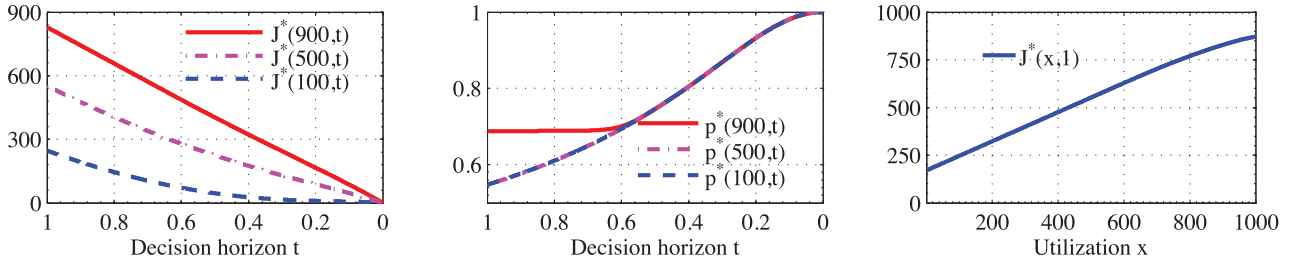
To summarize, we find that information delay has direct impact on revenue with dynamic pricing, and the provider has financial incentives to develop a responsive and accurate management system to obtain real-time information about the resource utilization.

5.5 Nonhomogeneous Demand

Finally, we consider the nonhomogeneous demand model, where the demand arrival and departure rate is $X(s)f(p(s))$ and $X(s)g(p(s))$, respectively. We use $f(p) = \sqrt{1-p^2}$ and $g(p) = 1 - \sqrt{1-p^2}$. This is consistent with the homogeneous demand functions except for the constant k, which is equal to 1 for the nonhomogeneous case so that $f(\cdot)$ and $g(\cdot)$ represent probabilities. The same discretization technique is used to solve the dynamic program using backward induction, and the entire decision horizon is discretized into $N = 1{,}000$ intervals. The optimal price in this case can be readily obtained to be $p^*(x,t) = 1/\sqrt{1+y^2}$, where $y = J^*(x+1, t-\Delta t) - J^*(x-1, t-\Delta t)$.

We first evaluate a small system with $C = 100$. Fig. 8 shows the results. From Fig. 8a, we can see that the optimal revenue $J^*(x,t)$ is increasing in x and t, which shows that Theorem 5 for the asymptotic case also holds in general. The interesting story is in Fig. 8b. Compared to Figs. 6b and 7b, optimal price $p^*(x,t)$ is largely indifferent for different values of x when t is smaller than around 0.5. For $t > 0.5$ and $t < 0.3$, $p^*(x,t)$ is still distinct for different x. This demonstrates the intuition revealed by Theorem 6 that the optimal price becomes much less dependent on the utilization x, when demand is non-homogeneous.

We then consider a large system with $C = 1{,}000$, and Fig. 9 shows the results. The revenue result in Fig. 9a does



(a) Optimal revenue. (b) Optimal price. (c) Optimal revenue vs. utilization at $t = 1$.

Fig. 9. Numerical results with $C = 1{,}000$, $f(p) = \sqrt{1-p^2}$, $g(p) = 1 - \sqrt{1-p^2}$, and $N = 1{,}000$ time intervals.

not show much difference from the small capacity case. However, the optimal price becomes identical for different values of x when $t < 0.6$ as shown in Fig. 9b, and the difference is smaller when $t > 0.6$ compared to the small capacity case in Fig. 8b. This is consistent with Theorem 6 that says when the capacity is large, the optimal price becomes independent of the utilization x. Optimal revenue is linear in x as shown in Figs. 8c and 9c and expected from the fact that $M(x,t) = M(t)$ in the asymptotic case.

6 Discussions

In this section, we discuss several issues pertaining to the practicality and usefulness of this work, which also lead to possible directions of future work.

6.1 Benefits of Dynamic Pricing for Users

In this paper, we mainly focus on the provider. Nevertheless, dynamic pricing is also beneficial for users of the cloud. With static pricing, the provider has limited ways to control the user demand. When demand for resources (e.g., virtual machines, bandwidth, etc.) increases, the performance of the virtual machines degrades and the probability of failures increases, leading to inferior user experience. With dynamic pricing, the provider has an effective means to dynamically control the demand, and ensure the overall performance of the cloud is satisfactory for customers. Thus, we believe that dynamic pricing is also beneficial to users, from the performance point of view.

We did not explicitly model the resource contention (e.g., bandwidth, CPU, memory) and its effect on user experience of using the cloud, which is an interesting future direction of extending the work. Such an effect is in fact an important topic in our community. Recently, there has been active research on providing bandwidth guarantees and different notions of fairness to users of the cloud (for example, [17], [22], [23], [35]).

We have also reached out to public cloud providers such as Microsoft Azure to comment on the practicality of dynamic pricing [7]. They view dynamic pricing as an viable option that will be increasingly adopted in addition to static pricing. The motivation is that providers would like to accommodate various customers to increase our revenue: some customers care more on guaranteed SLA with fixed costs so they may still choose static price, while some have the flexibility to trade response time/quality for their cost with dynamic pricing, by using less resources when prices are high [7].

6.2 Potential Concerns

Here, we also discuss some limitations of dynamic pricing and potential concerns a provider may have. First, as just mentioned, dynamic pricing is more suitable for flexible workloads that can be stopped or adjusted during the execution. For example, if the workload is batch data processing, it can be postponed to times when the price is cheaper, or the user can purchase less virtual machines to run it when price is high. However, if the workload is a web service that has to response to user requests, then the user has very little flexibility to adjust the consumption according to the price. This may prevent the adoption especially for smaller providers with few flexible workloads.

Second, to implement dynamic pricing, the provider needs to collect demand information and empirically derive the demand arrival and departure rate functions. It may introduce a trial period, before the actual adoption, to accumulate enough demand data to calculate the expected demand change corresponding to a price change. Demand information also needs to be continuously monitored and analyzed to refine the functions and pricing strategy. This may add managerial overhead to the provider.

Finally, for users, dynamic pricing may seem complicated to understand psychologically, and they may prefer fixed pricing for simplicity.

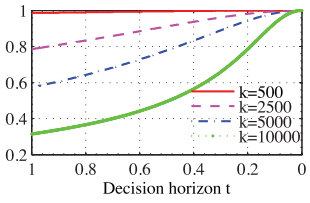


Fig. 10. Optimal price $p^*(5{,}000, t)$ for different values of k. $C = 10{,}000$, and $N = 10{,}000$ time intervals.

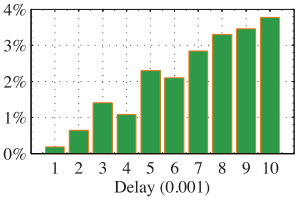


Fig. 11. Revenue loss due to delay for a period of $[0{,}0.25]$. $C = 10{,}000$, $k = 10{,}000$, $x = 5{,}000$, $N = 1{,}000$ intervals.

6.3 Alternative Pricing Strategies

We studied dynamic pricing from a revenue management perspective, which is widely adopted in traditional industries such as airlines, car rental, and hotels. Public cloud providers share more similarity with Internet companies such as Google and Facebook. Here, we briefly survey the pricing strategies used by Internet companies and comment on their applicability to cloud computing.

Internet companies usually provide their services free of charge, and generate revenues from selling ads online. Examples include Google AdWords [6] and Facebook Ads [1]. Since the values of ads for different goods and services could vary significantly and are difficult to determine, bidding/auctions are usually used as the pricing strategy [6]. Users submit bids as the maximum price they are willing to pay for each click of their ads, and the company runs an auction to determine which ads are displayed on their page, and their rank on the page [6].

Compared to bidding, dynamic pricing has the following advantages that are briefly discussed in Section 2.2. First, with dynamic pricing, the provider has more control over the price, while with bidding the provider cannot control the final sale price, which can be well below the break-even point for the provider to recover its costs. Second, dynamic pricing eliminates the potential user collusion, which is a common and difficult problem in practice for auctions [30]. With bidding, a group of users can collaborate to game the mechanism for their own benefits in various ways. A public cloud has a large number of users, making collusion detection and prevention even more difficult.

Some may argue that users are unfairly taken advantage of with dynamic pricing compared to bidding, as the price is solely decided by the provider. We emphasize that first price is determined taking into account users' reactions according to the demand arrival rate and departure rate functions as in Section 2.2. Moreover, competition in the market provides choices for users, and forces the price to remain at a reasonable level. This is an important topic by itself and can be explored further in future work.

6.4 Costs and Profit Maximization

We have mainly discussed revenue maximization without considering costs. Operating costs of a cloud, mostly power draw of servers and cooling systems, are fixed and do not vary with the infrastructure utilization. The reason is that operators typically leave all the servers and switches on at all times [18]. Thus, we did not consider costs here.

Recently, some studies propose to dynamically turn off/on servers and/or switches according to the workload to save energy [26]. In these cases, costs are proportional to demand, and a profit maximization formulation is more appropriate for studying pricing which is beyond the scope of this paper. Readers are referred to our work in [44] for more details.

7 Related Work

Our work has roots in revenue management. Since the seminal work of [15], dynamic pricing has become an active topic of revenue management, with many successful real-world applications (see [14] and references therein). As discussed in Section 1, with cloud computing, the unique challenge is that we need to model stochastic demand departures in addition to demand arrivals. This is because in our problem, price is charged per unit of time and sale, while previous works only consider the simple case of charging per unit of sale.

An extensive literature exists on pricing in communication networks and Internet. Many works study offline pricing, where pricing is computed offline to optimize revenue, social welfare, and so on. Odlyzko [32] argues that the predominant flat-rate pricing for selling retail Internet access encourages waste and is incompatible with service differentiation. The benefits of usage-based pricing are studied in [25], [37], where it is shown that with price differentiation one can use resources more efficiently. Paris Metro Pricing, in which service differentiation and congestion control are autonomously achieved by charging different prices for different service tiers that share the same infrastructure, is thoroughly studied in [11], [12], [31]. In [40], tiered pricing for Internet transit is further studied. Time is another dimension to unbundle connectivity. Hande et al. [20] characterize the economic loss due to the ISP's inability or unwillingness to price broadband access based on time of day. Ha et al. [19] study the time-dependent pricing for mobile data.

Our work is more related to the *online pricing* literature that deals with instantaneous demand dynamics and adjusts price on the spot. Compared with offline pricing it is less explored in the community. Paschalidis and Liu [33], Paschalidis and Tsitsiklis [34] study online pricing based on congestion in networks as a dynamic program, and show that static pricing achieves good performance in large networks. Our main concern in this paper is revenue maximization using dynamic pricing, and we consider a finite horizon formulation, which is different from the infinite horizon setting in [33], [34].

There have been some recent studies on the market and pricing of cloud resources. Javadi et al. [21] propose a stochastic model for the spot prices of EC2. The focus of [21] is to better model and predict spot prices, while our focus is to develop a new dynamic pricing scheme that improves the revenue of the operator. Wang et al. [42] argue for the importance of pricing in cloud computing for distributed systems design. From a user perspective, Teng and Magoulès [39] study the equilibrium pricing and allocation policy between multiple users using game theory. From a provider perspective, Mihailescu and Teo [29] propose a computationally efficient pricing scheme based on mechanism design, and Macías and Guitart [28] adopt a genetic algorithm to iteratively optimize the pricing policy. The most related work to ours are [24] and our own work [44], where pricing strategies are developed by solving some optimization problems. These approaches are primarily of a one-shot nature without considering the effect of pricing on future demand and revenue.

8 Concluding Remarks

We presented a revenue maximization framework to tackle the dynamic pricing problem in an IaaS cloud. The unique challenge is that prices are charged per instance per time

unit, and as a result the demand departure process has to be explicitly modeled. A stochastic dynamic program is formulated, and optimality conditions and important structural results of the optimal pricing policies are presented. We then extended to a general nonhomogeneous demand model.

We wish to emphasize that the broad literature of revenue management provides many meaningful future directions to study cloud resource pricing. For example, we can model the resupply of computing resources to the problem, which corresponds to the inventory control aspect of the framework. Since different kinds of resources are involved, how to choose the optimal mix of resources to create a menu of final products, such as Amazon's various instance types, is also an important issue.

REFERENCES

[1] *Advertising on Facebook,* https://www.facebook.com/about/ads/, 2013.

[2] *Amazon EC2,* http://aws.amazon.com/ec2/. 2013.

[3] *Amazon EC2 API Tools,* http://aws.amazon.com/developertools/351, 2013.

[4] *Amazon EC2 Spot Instances,* http://aws.amazon.com/ec2/spot-instances, 2013.

[5] *Amazon's EC2 Generating 220M+ Annually,* http://cloudscaling.com/blog/cloud-computing/amazons-ec2-generating-220m-annually, 2013.

[6] *How Costs Are Calculated in AdWords,* https://support.google.com/adwords/answer/1704424, 2013.

[7] Personal Communication with Yuxiong He, Microsoft Research, Sept. 2013.

[8] O.A. Ben-Yehuda, M. Ben-Yehuda, A. Schuster, and D. Tsafrir, "Deconstructing Amazon EC2 Spot Instance Pricing," *Proc. IEEE Third Int'l Conf. Cloud Computing Technology and Science (CloudCom),* 2011.

[9] G. Bitran and R. Caldentey, "An Overview of Pricing Models for Revenue Management," *Manufacturing & Service Operations Management,* vol. 5, no. 3, pp. 203-229, Sept. 2003.

[10] P. Bremaud, *Point Processes and Queues, Margingale Dynamics.* Springer-Verlag, 1980.

[11] X.R. Cao, H.X Shen, R. Milito, and P. Wirth, "Internet Pricing with a Game Theoretical Approach: Concepts and Examples," *IEEE/ACM Trans. Networking,* vol. 10, no. 2, pp. 208-216, Apr. 2002.

[12] C.-K. Chau, Q. Wang, and D.-M. Chiu, "On the Viability of Paris Metro Pricing for Communication and Service Networks," *Proc. IEEE INFOCOM,* 2010.

[13] J. Chen, C. Wang, B.B. Zhou, L. Sun, Y.C. Lee, and A.Y. Zomaya, "Tradeoffs between Profit and Customer Satisfaction for Service Provisioning in the Cloud," *Proc. 20th Int'l Symp. High Performance Distributed Computing (HPDC),* 2011.

[14] W. Elmaghraby and P. Keskinocak, "Dynamic Pricing in the Presence of Inventory Considerations: Research Overview, Current Practices, and Future Directions," *Management Science,* vol. 49, no. 10, pp. 1287-1309, Oct. 2003.

[15] G. Gallego and G. van Ryzin, "Optimal Dynamic Pricing of Inventories with Stochastic Demand over Finite Horizons," *Management Science,* vol. 40, no. 8, pp. 999-1020, Aug. 1994.

[16] M.K Geraghty and E. Johnson, "Revenue Management Saves National Car Rental," *Interfaces,* vol. 27, no. 1, pp. 107-127, 1997.

[17] A. Ghodsi, M. Zaharia, B. Hindman, A. Konwinski, S. Shenker, and I. Stoica, "Dominant Resource Fairness: Fair Allocation of Multiple Resource Types," *Proc. USENIX Conf. Networked Systems Design and Implementation (NSDI),* 2011.

[18] A. Greenberg, J. Hamilton, D.A. Maltz, and P. Patel, "The Cost of a Cloud: Research Problems in Data Center Networks," *ACM SIGCOMM Computer Comm. Rev.,* vol. 39, no. 1, pp. 68-73, 2009.

[19] S. Ha, S. Sen, C. Joe-Wong, Y. Im, and M. Chiang, "TUBE: Time Dependent Pricing for Mobile Data," *Proc. Proc. ACM SIGCOMM,* 2012.

[20] P. Hande, M. Chiang, R. Calderbank, and J. Zhang, "Pricing under Constraints in Access Networks: Revenue Maximization and Congestion Management," *Proc. IEEE INFOCOM,* 2010.

[21] B. Javadi, R. Thulasiramy, and R. Buyya, "Statistical Modeling of Spot Instance Prices in Public Cloud Environments," *Proc. IEEE Int'l Conf. Utility and Cloud Computing (UCC),* 2011.

[22] V. Jeyakumar, M. Alizadeh, D. Mazieres, B. Prabhakar, C. Kim, and A. Greenberg, "EyeQ: Practical Network Performance Isolation at the Edge," *Proc. USENIX Conf. Networked Systems Design and Implementation (NSDI),* 2013.

[23] C. Joe-Wang, S. Sen, T. Lan, and M. Chiang, "Multi-Resource Allocation: Fairness-Efficiency Tradeoffs in a Unifying Framework," *Proc. IEEE INFOCOM,* 2012.

[24] V. Kantere, D. Dash, G. Francois, S. Kyriakopoulou, and A. Ailamaki, "Optimal Service Pricing for a Cloud Cache," *IEEE Trans. Knowledge and Data Eng.,* vol. 23, no. 9, pp. 1345-1358, Feb. 2011.

[25] G. Kesidis, A. Das, and G. de Veciana, "On Flat-Rate and Usage-Based Pricing for Tiered Commodity Internet Services," *Proc. 42nd Ann. Conf. Information Sciences and Systems (CISS),* 2008.

[26] M. Lin, A. Wierman, L.L.H. Andrew, and E. Thereska, "Dynamic Right-Sizing for Power-Proportional Data Centers," *Proc. IEEE INFOCOM,* 2011.

[27] T. Lossen, "Cloud Exchange," http://cloudexchange.org/, 2013.

[28] M. Macías and J. Guitart, "A Genetic Model for Pricing in Cloud Computing Markets," *Proc. Symp. Applied Computing,* 2011.

[29] M. Mihailescu and Y.M. Teo, "On Economic and Computational-Efficient Resource Pricing in Large Distributed Systems," *Proc. 10th IEEE/ACM Int'l Symp. Cluster, Cloud and Grid Computing,* 2010.

[30] P. Milgrom, *Putting Auction Theory to Work.* Cambridge Univ. Press, 2004.

[31] A. Odlyzko, "Paris Metro Pricing for the Internet," *Proc. ACM First ACM Conf. Electronic Commerce (EC),* 1999.

[32] A. Odlyzko, "Should Flat-Rate Internet Pricing Continue?" *IEEE IT Professional,* vol. 2, no. 5, pp. 48-51, Sept. 2000.

[33] I.C. Paschalidis and Y. Liu, "Pricing in Multiservice Loss Networks: Static Pricing, Asymptotic Optimality, and Demand Substitution Effects," *IEEE/ACM Trans. Networking,* vol. 10, no. 3, pp. 425-438, June 2002.

[34] I.C. Paschalidis and J.N. Tsitsiklis, "Congestion-Dependent Pricing of Network Services," *IEEE/ACM Trans. Networking,* vol. 8, no. 2, pp. 171-184, Apr. 2000.

[35] L. Popa, G. Kumar, M. Chowdhury, A. Krishnamurthy, S. Ratnasamy, and I. Stoica, "Faircloud: Sharing the Network in Cloud Computing," *Proc. ACM SIGCOMM,* 2012.

[36] P.V Schaeffer, *Commodity Modeling and Pricing: Methods for Analyzing Resource Market Behavior.* John Wiley & Sons, 2008.

[37] S. Shakkottai, R. Srikant, A. Ozdaglar, and D. Acemoglu, "The Price of Simplicity," *IEEE J. Selected Areas in Comm.,* vol. 26, no. 7, pp. 1269-1276, Sept. 2008.

[38] B. Smith, R.J Leimkuhler, and J.S Darrow, "Yield Management at American Airlines," *Interfaces,* vol. 22, pp. 8-31, 1992.

[39] F. Teng and F. Magoulès, "Resource Pricing and Equilibrium Allocation Policy in Cloud Computing," *Proc. 10th IEEE Int'l Conf. Computer and Information Technology (CIT '10),* 2010.

[40] V. Valancius, C. Lumezanu, N. Feamster, R. Johari, and V.V. Vazirani, "How Many Tiers? Pricing in the Internet Transit Market," *Proc. ACM SIGCOMM,* 2011.

[41] K. Vermeersch, "Spot Watch," http://spotwatch.eu/input/, 2013.

[42] H. Wang, Q. Jing, R. Chen, B. He, Z. Qian, and L. Zhou, "Distributed Systems Meet Economics: Pricing in the Cloud," *Proc. Second USENIX Conf. Hot Topics in Cloud Computing (HotCloud),* 2010.

[43] L. Weatherford and S. Bodily, "A Taxonomy and Research Overview of Perishable-Asset Revenue Management: Yield Management, Overbooking, and Pricing," *Operations Research,* vol. 40, no. 5, pp. 831-844, Sept. 1992.

[44] H. Xu and B. Li, "A Study of Pricing for Cloud Resources," *ACM SIGMETRICS Performance Evaluation Rev., Special Issue on Cloud Computing,* vol. 40, no. 4, pp. 3-12, Mar. 2013.

[45] Q. Zhang, E. Gurses, R. Boutaba, and J. Xiao, "Dynamic Resource Allocation for Spot Markets in Clouds," *Proc. 11th USENIX Conf. Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services (Hot-ICE),* 2011.

Hong Xu received the BEngr degree from the Department of Information Engineering, The Chinese University of Hong Kong in 2007 and the MASc and PhD degrees from the Department of Electrical and Computer Engineering, University of Toronto, in 2010 and 2013, respectively. He is currently an assistant professor in the Department of Computer Science, City University of Hong Kong. His research interests include the design, analysis, and implementation of large-scale networked systems in cloud computing, data centers, and wireless networks. He is a member of the IEEE, IEEE Computer Society, and ACM.

Baochun Li received the BEngr degree from the Department of Computer Science and Technology, Tsinghua University, China, in 1995 and the MS and PhD degrees from the Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, in 1997 and 2000, respectively. Since 2000, he has been with the Department of Electrical and Computer Engineering at the University of Toronto, where he is currently a professor. He holds the Nortel Networks junior chair in Network Architecture and Services from October 2003 to June 2005, and the Bell Canada Endowed chair in Computer Engineering since August 2005. His research interests include large-scale distributed systems, cloud computing, peer-to-peer networks, applications of network coding, and wireless networks. He received the IEEE Communications Society Leonard G. Abraham Award in the field of communications systems in 2000. In 2009, he received the Multimedia Communications Best Paper Award from the IEEE Communications Society, and the University of Toronto McLean Award. He is a senior member of the IEEE and IEEE Computer Society and a member of ACM.