Game Theory-Based Nonlinear Bandwidth Pricing for Congestion Control in Cloud Networks

Abouzar Ghavami*, Zhuozhao Li† and Haiying Shen†
Department of Electrical and Computer Engineering, Clemson University, Clemson, SC 29631
†Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22904

Email: ghavamip@gmail.com, zl5uq@virginia.edu, hs6ms@virginia.edu

***Abstract*—In the cloud, the network links are shared among tenants, which makes them easy to get fully congested (overloaded). Overloaded links degrade the performance of tenants' applications, and impose additional costs to the cloud provider. In this paper, we propose a nonlinear bandwidth pricing policy for congestion control in the cloud network. In order to maximize social welfare (i.e., maximize the total satisfaction of the tenants while minimizing the congestion over the link), the cloud provider uses the nonlinear pricing policy that increases the unit price with increment of bandwidth usage. Each tenant competes for bandwidth allocation to maximize its utility (i.e., both maximize its own individual satisfaction and minimize its bandwidth payment cost). We design a game between tenants and the cloud provider, and show that there exists a unique optimal bandwidth schedule (Nash equilibrium) that jointly maximizes the social welfare and the utility of each tenant at the same time. In order to find the optimal schedule, we use an asynchronous-based best response strategy, in which each tenant updates its optimal bandwidth allocation based on the updated bandwidth payment function from the cloud provider. We prove that the updated bandwidth allocations converge to the optimal bandwidth schedule. In our simulation study and real implementation, we verify the performance of our proposed pricing mechanism under different scenarios.**

I. Introduction

Many of today's web services (e.g., Dropbox and Netflix) are deployed on the cloud. The cloud providers own and maintain the large-scale computation, storage and network resources, and the tenants rent the resources on demand instead of purchasing the costly servers, software, datacenter space or network equipments [1]. Many web services such as transaction processing web applications [2] and video-on-demand (VoD) applications require predictable network performance to ensure the performance of their end-users. Thus, it is important for the cloud providers to guarantee the quality of service (QoS) of the tenants in order to attract and maintain tenants, and hence earn profit by keeping a steady long-term business.

Recently, many bandwidth allocation policies have been proposed [3]–[8]. Optimal bandwidth allocation is addressed extensively in the previous studies. Kelly [9] and Low [10] propose decentralized optimization methods using linear pricing to maximize the total satisfaction of users in a network with limited capacity links. Niu *et al.* [3] proposed a linear pricing model to jointly maximize the total satisfaction of the tenants and minimize the reservation payment cost of the cloud provider. The authors found the optimal bandwidth reservation through two different distributed and step-size-free optimization methods, Chaotic price update and cutting-plane method. However, by charging tenants at a flat rate for bandwidth usage, the above works do not aim to control congestion and avoid overloading on the links, which is very essential for guaranteeing the QoS of the tenants. Our work allows the cloud provider to control congestion and avoid overloading in the cloud network.

In this paper, we maximize the social welfare of the tenants that also minimizes the network congestion in addition to maximize the total satisfaction of the tenants in the network. Our bandwidth payment method is also more transparent to the tenants than the proposed methods in [9] and [10], where the bandwidth payment for each tenant is updated based on the payment rate at the previous update step and the total bandwidth consumption of the tenants. In our proposed method, the pricing mechanism at each iteration step is evaluated based on a pre-determined congestion cost function, that is known for all the tenants before the first step of bandwidth update process, and it does not change during the update process. The bandwidth cost function is also determined based on the current total load in the network that is observable by the tenants. Furthermore, our proposed method provides more generalized nonlinear pricing functions for the cloud provider to control congestion in the network rather than linear pricing rates. We have also extended our proposed non-linear pricing method to multiple switch cores in [11]. The nomenclature of the notations used in this paper is summarized in table I.

The rest of this paper is organized as follows. We present the preliminaries overview in Section II. In Section III, we formulate the problem to maximize the social welfare and in Section IV, we propose the pricing policy that effectively avoids congestion and maximizes tenants' utility. In Section V, we define a game between the tenants, and we investigate the Nash equilibrium properties. We find the optimal bandwidth schedule using the best response strategy in Section VI, and we discuss the total payments of the tenants in Section VII. In Section VIII, we present our simulation and real implementation results to verify the performance of our proposed pricing mechanism. We conclude our paper with remarks on our future work in Section IX.

TABLE I: Nomenclature of the notations used in this paper

b_k	bandwidth of tenant k
$\mathcal{U}_k(\cdot)$	Satisfaction function of tenant k
$\mathcal{V}(\cdot)$	The congestion cost function
$\pi(\cdot)$	The unit bandwidth price at a specific bandwidth usage
l	Link l
$\mathbf{b}$	The bandwidth schedule of the tenants
$\mathcal{B}$	The feasible set of all bandwidth schedules
$\mathcal{B}_l$	Total bandwidth of the tenants on link l
$\mathcal{B}_l^{\max}$	Total bandwidth capacity on link l
η_l	Allowed link capacity percentage
$\mathcal{C}_l$	Overload cost function on link l
$\mathcal{Z}(\cdot)$	Total overload and congestion cost function on link l
$\mathcal{S}(\cdot)$	Social welfare of the tenants
$\psi_k(\cdot)$	Total payment of tenant k
$\mathcal{F}_k(\cdot, \psi_k(\cdot))$	Utility function of tenant k
$\mathcal{R}(\cdot)$	Revenue of the cloud provider

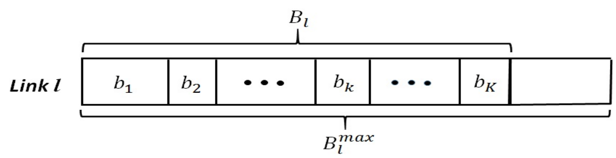


Fig. 1: The bandwidth usage of tenants in link l.

II. Preliminaries and Overview

Based on recent works on sharing data center networks [12], [13], we abstract the connections of tenants into a hose model [6], [14], where the network throughput of each tenant is blocked by a shared access link (denoted by link l). We consider a set of tenants, $\mathcal{K} = \{1, ..., K\}$, that use link l to send data to their destinations in the network. As shown in Fig. 1, $B_l^{\max}$ denotes the bandwidth capacity of link l, b_k denotes the amount of bandwidth that tenant k allocates for sending data through link l, and $B_l = \sum_{k\in\mathcal{K}} b_k$ denotes the total bandwidth on the link. We denote $\frac{B_l}{B_l^{\max}}$ as the *congestion degree* of link l.

Let b_k denote the amount of bandwidth allocated to tenant k on link l. Let us denote $\mathbf{b} = (b_1, ..., b_K)$ as the bandwidth schedule for the tenants on link l, and denote $\mathbf{b}_{-k}$ as the bandwidth schedule for all the tenants excluding tenant k on link l. A tenant may have a maximum required amount of bandwidth if it is not willing to pay more than a specific maximum bandwidth demand based on its application usage. Thus, we assume that each tenant k allocates a maximum amount of bandwidth denoted by $b_k^{\max}$, i.e. $b_k \le b_k^{\max}$. We have $b_k^{\max} = B_l^{\max}$ for a tenant without maximum bandwidth allocation. Let us denote $\mathcal{B}_k = [0, b_k^{\max}]$, and denote $\mathcal{B} = \mathcal{B}_1 \times ... \times \mathcal{B}_K$ as the set of all feasible bandwidth allocations of the tenants. The set $\mathcal{B}$ is a compact and convex set. The cloud provider defines a safety margin $\eta_l \le 1$, and tries to constrain the bandwidth consumption on the link to no more than $\eta_l B_l^{\max}$. A feasible bandwidth schedule must satisfy the link's capacity constraint:

$$B_l = \sum_{k\in\mathcal{K}} b_k \le \eta_l B_l^{\max}. \tag{1}$$

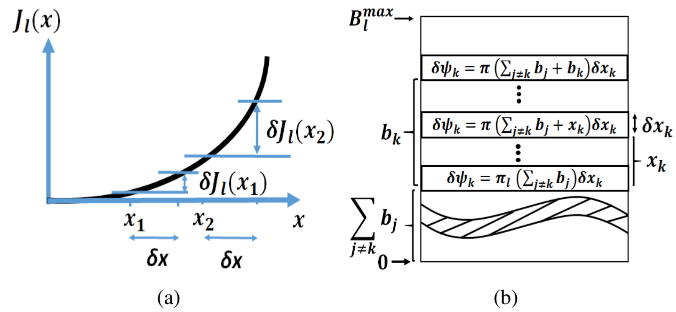


Fig. 2: (a) Congestion cost increments at different bandwidth usages and (b) Congestion-based bandwidth pricing.

III. Problem Formulation

In this section, we formulate the problem to find the optimal bandwidth schedule that satisfies tenants' demands as much as possible and avoids congestion in link l. To achieve this objective, the cloud provider specifies a nonlinear pricing policy that has a higher unit price when the current bandwidth usage is higher. This disincentivizes the tenants to use bandwidth when the bandwidth usage ($\mathcal{B}_l$) is higher (i.e., the link is more likely to be congested). It is possible for the cloud provider to determine the bandwidth payment according to a strictly convex function of the bandwidth usage. We refer this cost as *congestion cost*, and denote it as $\mathcal{V}(\cdot)$.

Let us denote $\pi = \mathcal{V}'$ as the derivative of the pricing function $\mathcal{V}$. Figure 2(a) shows the variations of the strictly convex congestion cost function as the bandwidth usage varies. It is seen that the congestion cost function is higher at the higher bandwidth usage. Figure 2(b) shows the increments of the bandwidth pricing function for user k (denoted as ψ_k) using the derivative of the congestion cost function, π. The cloud provider needs to transfer this congestion cost to tenants by charging them on their individual bandwidth usages. When the current bandwidth usage is at a higher level, tenants are more disincentivized from allocating more bandwidths, thus avoiding the link to be fully congested.

Let us denote $\mathcal{U}_k(b_k)$ as the satisfaction function of tenant k from using b_k amount of bandwidth. The satisfaction function, $\mathcal{U}_k(\cdot)$, is considered to be non-decreasing as each tenant desires high quality of service and a higher bandwidth provision makes a tenant more satisfied [3], [15]. Also, the marginal satisfaction of a user is non-increasing because a tenant's level of satisfaction gradually gets saturated when the provisioned bandwidth increases [3], [15]. Therefore, we consider that $\mathcal{U}_k(\cdot)$ is a strictly increasing and strictly concave function, and its second derivative is continuous in $\mathcal{B}_k$. In order to attract and maintain long-term business, the cloud provider needs to provide bandwidth to meet each tenant's satisfaction, and also provide an un-congested network support for all the tenants at the same time. Since these two factors affect the QoS to the tenants' applications, which represents their welfare, we define *social welfare* of tenants as a joint consideration of these two factors in the following:

$$\mathcal{P}(\mathbf{b}) = \sum_{k=1}^{K} \mathcal{U}_k(b_k) - \mathcal{V}\left(\sum_{k\in\mathcal{K}} b_k\right)$$
$$\text{s.t.} \quad B_l - \eta_l B_l^{\max} \leq 0,$$
$$\mathbf{b} \in \mathcal{B}. \tag{2}$$

The above equation considers the bandwidth capacity constraint as in (1). It also considers the maximum demand constraint for each tenant by constraint $\mathbf{b} \in \mathcal{B}$, i.e., $b_k \in [0, b_k^{\max}]$. Then, the cloud provider aims to find the bandwidth schedule that maximizes the social welfare, which is the objective of this paper.

To find the bandwidth schedule that maximizes the social welfare, we transform (2) to another objective that integrates the constraint in (2) [16]. When the total bandwidth demands $B_l > \eta_l B_l^{\max}$, link l is considered as overloaded. In this case, some bandwidth demands cannot be fully satisfied, which may lead to SLA penalty, reputation degradation and business loss to the cloud provider.

We call these costs caused by the overloaded link *overload cost*. We also let $\mathcal{C}(x)$ denote the overload cost function associated with link l. We define $\mathcal{C}(x)$ as a strictly convex function with $\mathcal{C}(x) = 0$ for $x < 0$. Let us denote

$$\mathcal{Z}(x) = \mathcal{V}(x) + \mathcal{C}(x - \eta_l B_l^{\max}) \tag{3}$$

as the total cost function and overload penalty function of consuming x unit of bandwidth. The social welfare of the tenants in (2) is rewritten as in the following:

$$\mathcal{S}(\mathbf{b}) = \sum_{k=1}^{K} \mathcal{U}_k(b_k) - \mathcal{Z}\left(\sum_{k\in\mathcal{K}} b_k\right),$$
$$\mathbf{b} \in \mathcal{B}. \tag{4}$$

As $\mathcal{U}_k(\cdot)$ is strictly concave for each tenant k, and $\mathcal{V}(\cdot)$ and $\mathcal{C}(\cdot)$ are strictly convex function, $\mathcal{S}(\cdot)$ is a strictly concave function in $\mathcal{B}$.

Definition 1. *A feasible bandwidth schedule is a socially optimal bandwidth schedule if it maximizes the social welfare of the tenants as in (4).*

Note that in the case that cloud provider knows satisfaction functions and maximum bandwidth requirements of the tenants, the optimization problem in (4) is solved using a single step standard convex optimization method [16]. In this paper, we consider that the tenants do not reveal their private information such as satisfaction function and maximum bandwidth requirements. In this case, the socially optimal bandwidth schedule is only derived through an iterative decentralized bandwidth allocation process that is described in the next following sections.

Below, we show how the cloud provider transfers the congestion cost to the tenants by charging each tenant based on its contribution to the congestion cost and overload cost, and in Section VI, we propose our decentralized optimization framework to find the socially optimal bandwidth allocation schedule.

IV. Proposed Pricing Policy

We now introduce how the cloud provider makes pricing policy to transfer the congestion cost and overload cost to the tenants in order to incentivize the tenants to voluntarily constrain the costs. Let $\psi_k(x_k)$ denote the bandwidth payment function that tenant k pays for x_k amount of bandwidth reservation for the next time period T_i. The total payment of tenant k for reserving b_k amount of bandwidth, $\psi_k(b_k)$, is calculated as in the following:

$$\psi_k(b_k) = \mathcal{Z}\left(\sum_{j\in\mathcal{K}-\{k\}} b_j + b_k\right) - \mathcal{Z}\left(\sum_{j\in\mathcal{K}-\{k\}} b_j\right). \tag{5}$$

Note that the term $-\mathcal{Z}(\cdot)$ in (5) is independent from b_k. These terms result in an unbiased cost function for the tenants, i.e. $\psi_k(0) = 0$, $\forall k$.

Next, we introduce the utility of tenant k that it tries to maximize when determining its bandwidth demand. The gain and the cost of tenant k by using b_k amount of bandwidth can be measured by its satisfaction $\mathcal{U}_k(b_k)$ and bandwidth payment function, $\psi_k(\cdot)$, respectively. Thus, the utility function of tenant k, $\mathcal{F}_k(b_k, \psi_k(\cdot))$, for using b_k amount of bandwidth is calculated as in the following:

$$\mathcal{F}_k(b_k, \psi_k(\cdot)) = \mathcal{U}_k(b_k) - \psi_k(b_k), \quad b_k \in \mathcal{B}_k. \tag{6}$$

Based on the proposed bandwidth payment function, $\psi_k(\cdot)$ in (5), the utility function of tenant k, $\mathcal{F}_k(b_k, \psi_k(\cdot))$, is also a function of $\mathbf{b}_{-k}$. We refer $\mathcal{F}_k(b_k, \psi_k(\cdot))$ as a function of $\mathbf{b}_{-k}$ and b_k, and denote it as $\mathcal{F}_k(\mathbf{b}_{-k}, b_k)$. Therefore, the tenant utility function is calculated as in the following:

$$\mathcal{F}_k(\mathbf{b}_{-k}, b_k) = \mathcal{U}_k(b_k)$$
$$-\left(\mathcal{Z}\left(\sum_{j\in\mathcal{K}-\{k\}} b_j + b_k\right) - \mathcal{Z}\left(\sum_{j\in\mathcal{K}-\{k\}} b_j\right)\right), b_k \in \mathcal{B}_k. \tag{7}$$

In order to find the socially optimal bandwidth schedule, we define a strategic game between the tenants. In this game, the cloud provider proposes the bandwidth payment function $\psi_k(\cdot)$ to the tenants, and each tenant k responds its proposed strategy, i.e., its allocated bandwidth b_k, to the cloud provider. In the next following sections, we derive the Nash equilibrium properties of this game, where both the cloud provider and tenants gain the maximum utility. We show that Nash equilibrium is socially optimal and vice versa. We will also show that the best response strategy of the tenants converge to the socially optimal bandwidth schedule.

V. Properties of Nash Equilibrium

The bandwidth reservation amount chosen by rational tenants, who always try to maximize their individual profits as defined in (7), can be represented as a strategic game $\langle \mathcal{K}, \mathcal{B}_k, \mathcal{F}_k \rangle$. In this strategic game, the strategy of each tenant

k ($k \in \mathcal{K}$) is its bandwidth allocation, $b_k \in \mathcal{B}_k$. Tenant k chooses its strategy, b_k, as a response to the bandwidth payment function, $\psi_k(b_k)$, that is determined based on the banwidth allocations of other tenants sharing link l. The best reponse strategy of tenant k is the bandwidth allocation that maximizes its individual utility function, $\mathcal{F}_k(\cdot)$.

In this section, we investigate the Nash equilibirium of this strategic game. We show that the Nash equilibrium exists, and it is equal to the *unique* socially optimal bandwidth schedule. That is, each tenant only has one choice of his bandwidth reservation in order to maximize its utility.

A. Existence of Nash Equilibrium.

The following conditions hold for each tenant $k \in \mathcal{K}$:

1. The set $\mathcal{B}_k$ is a non-empty, convex, and compact subset of a finite-dimensional Euclidean space.

2. The tenant utility function, $\mathcal{F}_k(\mathbf{b}_{-k}, b_k)$, is continuous in $\mathbf{b}$ as $\mathcal{U}_k(\cdot)$, $\mathcal{V}(\cdot)$, and $\mathcal{C}(\cdot)$ are continuous.

3. The tenant utility function, $\mathcal{F}_k(\mathbf{b}_{-k}, b_k)$, is a strictly concave function of b_k as $\mathcal{U}_k(\cdot)$ is strictly concave, and $\mathcal{V}(\cdot)$ and $\mathcal{C}(\cdot)$ are strictly convex functions.

Then, based on the *Debreu, Glicksberg, Fan theorem* [17], the game $\langle \mathcal{K}, \mathcal{B}_k, \mathcal{F}_k \rangle$ has a pure strategy Nash equilibrium.

B. Social Optimality of Nash Equilibrium.

Theorem 1. *The socially optimal bandwidth schedule is a Nash equilibrium, and vice versa.*

Proof: Recall $\mathbf{b}^*$ denotes a socially optimal bandwidth schedule that maximizes the social welfare of the tenants as in (4). As $\mathcal{S}(\cdot)$ is strictly concave over $\mathcal{B}$, from the first order inequality condition [16], we have:

$$\nabla \mathcal{S}(\mathbf{b}^*)^T(\mathbf{b} - \mathbf{b}^*) = \sum_{k \in \mathcal{K}} \nabla_k \mathcal{F}_k(\mathbf{b}^*_{-k}, b^*_k)(b_k - b^*_k) \leq 0, \quad \forall \mathbf{b} \in \mathcal{B}. \tag{8}$$

where $\nabla_k \mathcal{F}_k = \frac{\partial \mathcal{F}_k}{\partial b_k}$ denotes the partial derivative of $\mathcal{F}_k$ with respect to b_k. Let us set $\mathbf{b} = (\mathbf{b}^*_{-k}, b_k)$ in (8), where $b_k \in \mathcal{B}_k$. Thus, for all $k \in \mathcal{K}$, we have:

$$\nabla_k \mathcal{F}_k(\mathbf{b}^*_{-k}, b^*_k)(b_k - b^*_k) \leq 0, \qquad \forall b_k \in \mathcal{B}_k. \tag{9}$$

As $\mathcal{F}_k$ is strictly concave over $\mathcal{B}_k$, from the above Proposition, the condition in (9) is also sufficient for b^*_k to maximize $\mathcal{F}_k(b_k)$ over $\mathcal{B}_k$, $\forall k$. Therefore, no tenant k deviates from its bandwidth reservation choice b^*_k, $\forall k$. Thus, the socially optimal bandwidth schedule $\mathbf{b}^*$ is a Nash equilibrium.

To prove the converse, let $\tilde{\mathbf{b}}$ is a Nash equilibrium bandwidth schedule that maximizes the utility functions of the tenants, and tenants do not deviate from it. As $\mathcal{F}_k\left(\tilde{\mathbf{b}}_{-k}, b_k\right)$ is strictly concave in $\mathcal{B}_k$, from the first order inequality condition, we have:

$$\nabla_k \mathcal{F}_k\left(\tilde{\mathbf{b}}_{-k}, b_k\right)\left(b_k - \tilde{b}_k\right) \leq 0, \qquad \forall k, b_k \in \mathcal{B}_k. \tag{10}$$

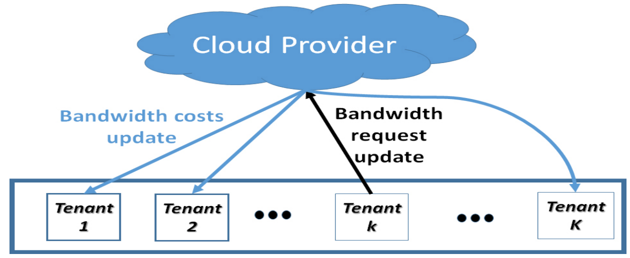


Fig. 3: The iterative bandwidth update process.

Writing the first order inequality condition for $\mathcal{S}$ in (4) at $\tilde{\mathbf{b}}$, and using (7) and the result in (10), for all $\mathbf{b} \in \mathcal{B}$, we have:

$$\nabla \mathcal{S}(\tilde{\mathbf{b}})^T\left(\mathbf{b} - \tilde{\mathbf{b}}\right) = \sum_{k \in \mathcal{K}} \nabla_k \mathcal{F}_k\left(\tilde{\mathbf{b}}_{-k}, b_k\right)\left(b_k - \tilde{b}_k\right) \leq 0. \tag{11}$$

As $\mathcal{S}(\cdot)$ is strictly concave in $\mathbf{b}$, from Proposition 2.1.2 (b) in [16], $\tilde{\mathbf{b}}$ maximizes $\mathcal{S}(\mathbf{b})$ over $\mathcal{B}$. Therefore, the Nash equilibrium bandwidth schedule, $\tilde{\mathbf{b}}$, is a socially optimal bandwidth schedule. ■

The following Lemma follows immediately from the strictly concavity of $\mathcal{S}(\cdot)$ in $\mathcal{B}$.

Lemma 1. *The socially optimal bandwidth schedule is* unique.

C. Uniqueness of the Nash Equilibrium

Theorem 2. *The Nash equilibrium bandwidth schedule is unique.*

Proof: The proof follows immediately from Theorem 1 and Lemma 1. ■

VI. Asynchronous-based Best Response Strategy

In this section, we find the socially optimal bandwidth schedule through a distributed method. As we mentioned, the tenants usually do not release their private information, such as satisfaction function, $\mathcal{U}_k(\cdot)$, and maximum bandwidth requirement, $b_k^{\max}(\cdot)$, to the cloud provider. Without knowing this information, the cloud provider is not able to find the socially optimal bandwidth schedule to maximize the social welfare as in (4) in a centralized manner. Then, to find the optimal bandwidth schedule, we propose a decentralized bandwidth allocation framework, in which the cloud provider uses an asynchronous-based best response strategy process [18] to allocate the bandwidth for the tenants.

A. Decentralized Bandwidth Allocation

Figure 3 shows this asynchronous-based best response strategy process. In each iteration step of this process, a random tenant updates its bandwidth allocation to maximize its utility ($\mathcal{F}_k(b_k, \psi_k(\cdot))$ in (6)) based on the bandwidth payment function ($\psi_k(\cdot)$ in (5)) from the cloud provider. Let $\mathbf{b}^m$ denote the updated bandwidth allocations of the tenants at iteration step m. As mentioned, we assume that the initial bandwidth demands of the tenants, $\mathbf{b}^0$, is predictable from historical data [19].

At the beginning of each step $m+1$, the cloud provider updates the bandwidth payment, $\psi_k^{m+1}(\cdot)$, for each tenant according to (5) based on the bandwidth allocations in the m^{th} step, $\mathbf{b}^m$, cost function $\mathcal{V}(\cdot)$, and the overload cost function $\mathcal{C}(\cdot)$ as in (12).

$$\psi_k^{m+1}(b_k) = \mathcal{Z}\left(\sum_{j\in\mathcal{K}-\{k\}} b_j^m + b_k\right) - \mathcal{Z}\left(\sum_{j\in\mathcal{K}-\{k\}} b_j^m\right). \tag{12}$$

Next, the cloud provider announces the new bandwidth payment functions, $\psi_k^{m+1}(\cdot)$, to the tenants. Each tenant updates its bandwidth allocation, b_k^{m+1}, to maximize its individual utility as in (13) and sends the updated bandwidth allocation to the cloud provider.

$$\begin{aligned} b_k^{m+1} &= \arg\max_{b_k\in\mathcal{B}_k} \mathcal{F}_k(\mathbf{b}_{-k}^m, b_k) \\ &= \arg\max_{b_k\in\mathcal{B}_k} \mathcal{U}_k\left(b_k\right) - \psi_k^{m+1}(b_k) \end{aligned} \tag{13}$$

We define *cycle* as the set of N successive updates that each tenant updates at least once in $N+K$ number of successive updates and call N *cycle length*. After N updates, if a tenant fails to update its bandwidth allocation, the cloud provider asks for its bandwidth update. If the tenant does not respond to the cloud provider's allocation, the cloud provider will drop it out of the update process and only allocates the link bandwidth among other tenants. This ensures that each tenant updates its bandwidth at least once in $N+K$ number of successive updates.

This iterative process repeats until a specific large number of updates has reached and each tenant has updated its demand in order to achieve a small convergence error to the Nash equilibrium bandwidth schedule. We show that the best response strategy of the tenants converges to the socially optimal bandwidth schedule, $\mathbf{b}^* = (b_1^*, ..., b_K^*)$, under this decentralized asynchronous updates.

The following Lemma follows from the strictly concavity of $\mathcal{F}_k(\mathbf{b}_{-k}, b_k)$ in $\mathcal{B}_k$.

Lemma 2. *At each update step, the optimal bandwidth allocation of the updating tenant is* unique.

B. Best Reponse Bandwidth Update of Tenants

In the proposed asynchronous-based best response update process, tenant k solves (13) to find its own optimal bandwidth allocation at step $m+1$. This optimal bandwidth allocation is derived as in the following Lemma.

Lemma 3. *The optimal bandwidth allocation of tenant k at $(m+1)^{th}$ update iteration is derived as in the following:*

$$b_k^{m+1} = \begin{cases} 0, & \nabla_k \mathcal{F}_k\left(\mathbf{b}_{-k}^m, 0\right) < 0, \\ b_k^{\max}, & \nabla_k \mathcal{F}_k\left(\mathbf{b}_{-k}^m, b_k^{\max}\right) > 0, \\ \arg_{b_k\in\mathcal{B}_k}\left\{\nabla_k \mathcal{F}_k\left(\mathbf{b}_{-k}^m, b_k\right) = 0\right\}, & \textit{O.W.} \end{cases} \tag{14}$$

Proof: The proof follows using the first order inequality condition:

$$\nabla_k \mathcal{F}_k\left(\mathbf{b}_{-k}^m, b_k^{m+1}\right)\left(b_k - b_k^{m+1}\right) \le 0, \quad \forall b_k \in \mathcal{B}_k. \tag{15}$$

for three possible cases: (i) $\nabla_k \mathcal{F}_k\left(\mathbf{b}_{-k}^m, 0\right) < 0$, (ii) $\nabla_k \mathcal{F}_k\left(\mathbf{b}_{-k}^m, b_k^{\max}\right) > 0$ and (iii) $\nabla_k \mathcal{F}_k\left(\mathbf{b}_{-k}^m, 0\right) \ge 0$ and $\nabla_k \mathcal{F}_k\left(\mathbf{b}_{-k}^m, b_k^{\max}\right) \le 0$. For these three possible cases, the unique solution to (13) that maximizes the utility of tenant k at $(m+1)^{th}$ update step is derived as in (14). ■

Given the bandwidth payment function, $\psi_k(\cdot)$, that is determined based on the congestion cost function $\mathcal{V}(\cdot)$, overload cost function, $\mathcal{C}(\cdot)$, and the bandwidth allocations of the tenants, $\mathbf{b}$, the tenants find the unique Nash equilibrium bandwidth schedule through the best response strategy. As derived in Section V, the achieved Nash equilibrium is the socially optimal bandwidth schedule that uniquely maximizes the social welfare of the tenants as in (4).

The following theorem follows from the strictly concavity of $\mathcal{S}(\cdot)$ and $\mathcal{U}(\cdot)$ by modifying the Gauss-Siedel algorithm in [20].

Theorem 3. *The asynchronous update process converges to the socially optimal bandwidth schedule.*

VII. Total bandwidth payments of the tenants

In this section, we evaluate the total bandwidth payments of the tenants based on the proposed pricing mechanism. We show that the total bandwidth payments of the tenants is maximized at the fully congested link, i.e. $B_l = B_l^{\max}$. We consider the link capacity constraint as in (1) with the safety margin factor set at $\eta_l = 1$. With satisfaction of the maximum link capacity constraint as in (1), each tenant k pays $\mathcal{R}_k(\mathbf{b}_{-k}, b_k)$ for using b_k amount of bandwidth:

$$\mathcal{R}_k\left(\mathbf{b}_{-k}, b_k\right) = \mathcal{V}\left(\sum_{j\in\mathcal{K}} b_j\right) - \mathcal{V}\left(\sum_{j\in\mathcal{K}/\{k\}} b_j\right). \tag{16}$$

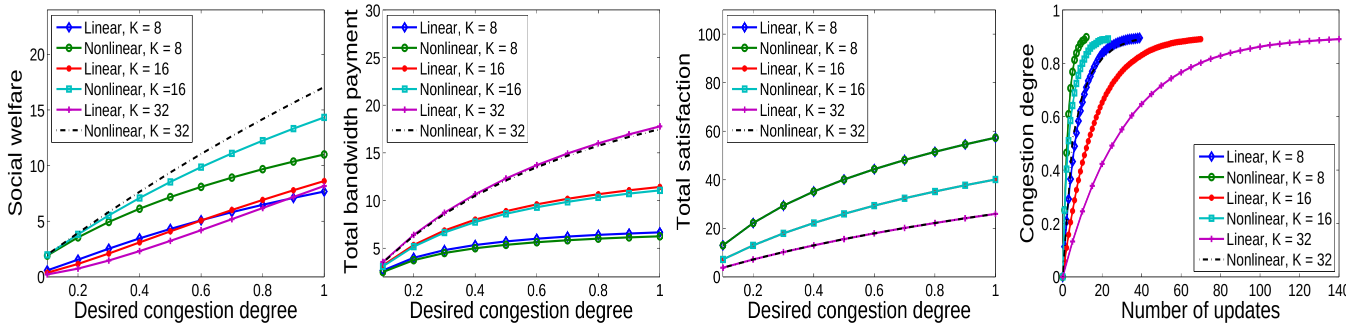
The total bandwidth payments of the tenants, $\mathcal{R}(\mathbf{b})$, is calculated as in the following:

$$\begin{aligned} \mathcal{R}(\mathbf{b}) &= \sum_{k\in\mathcal{K}} \mathcal{V}\left(\sum_{j\in\mathcal{K}} b_j\right) - \mathcal{V}\left(\sum_{j\in\mathcal{K}/\{k\}} b_j\right) \\ &= K\mathcal{V}\left(B_l\right) - \sum_{k\in\mathcal{K}} \mathcal{V}\left(B_l - b_k\right). \end{aligned} \tag{17}$$

From strictly convexity of $\mathcal{V}$, the total bandwidth payments of the tenants, $\mathcal{R}(\mathbf{b}_{-k}, \mathbf{b}_k)$, is strictly convex and increasing in b_k, $\forall k$. Thus, it gets maximized at $B_l = B_l^{\max}$.

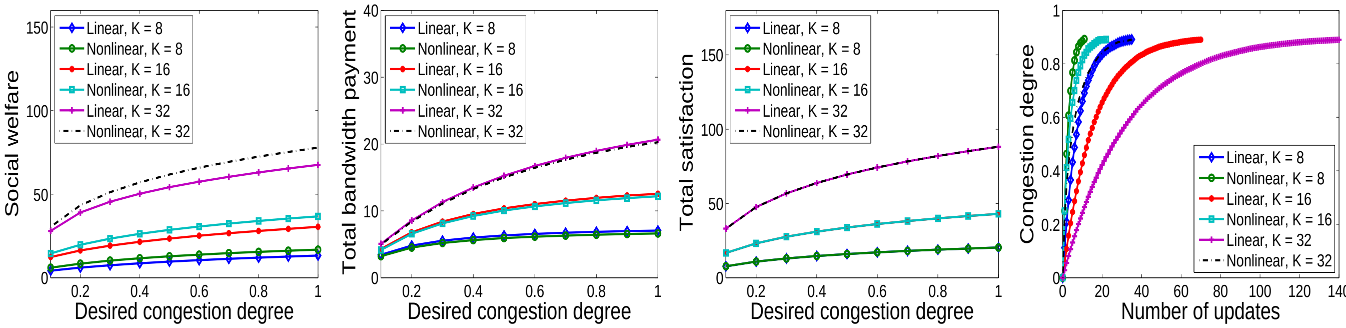
VIII. Performance Evaluation

In this section, we conduct simulations and real implementation experiments on Amazon EC2 [21] to evaluate the nonlinear pricing model compared with the linear pricing model.



(a) Social welfare (b) Total bandwidth payments (c) Total satisfaction (d) Congestion degree

Fig. 4: The simulation results of homogeneous tenants.



(a) Social welfare (b) Total bandwidth payments (c) Total satisfaction (d) Congestion degree

Fig. 5: The simulation results of heterogeneous tenants.

A. Environment of Simulation and Implementation

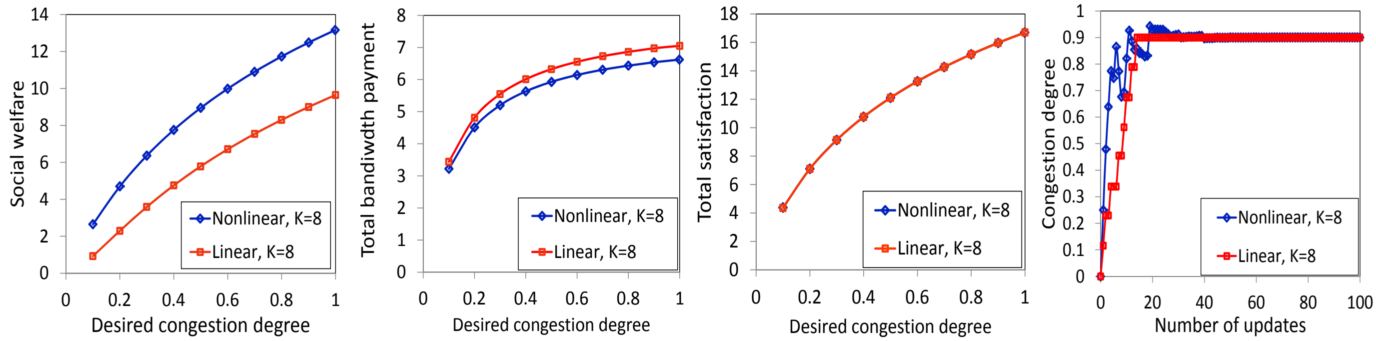
In both simulation and real implementation, we consider the following scenario. Assume there are K tenants and each tenant has 2 VMs connected with each other on the same link l. Each tenant has an application that sends data from one VM to another VMs through link l. Therefore, the K tenants compete the bandwidth resource in link l. We consider the strictly convex function $\mathcal{V}_1(x) = \alpha(\frac{x}{B_l^{\max}})$ as the linear congestion cost function, and $\mathcal{V}_2(x) = \beta(\frac{x}{B_l^{\max}})^2$ as the nonlinear congestion cost function in our simulation studies. The cloud provider determines the cost factors, β and α, to achieve a specified congestion degree on link l. The overload cost function was set to $\mathcal{C}(x) = \gamma x^2$ for $x > 0$, and $\mathcal{C}(x) = 0$ for $x \leq 0$. We set $\gamma = 100\beta$ and $\gamma = 100\alpha$ to make the overload cost large enough compared to the congestion cost function, $\mathcal{V}(x)$, for nonlinear and linear pricing policies, respectively. The capacity of link l is $B_l^{\max} = 40$ Gbps. In order to use the maximum link capacity, we set the safety margin factor $\eta_l = 1$. Unless otherwise specified, we set the number of updates in the best response strategy update process to 10000, which is large enough to converge to the optimal bandwidth schedule.

In our simulation, we tested the performance when the number of tenants set to $K = 8, 16$ and 32. We set the cycle length as $N = 8K$. We use $C_d = \frac{B_l}{B_l^{\max}}$ to denote the desired congestion degree that the cloud provider aims to achieve. We consider two different scenarios: i) homogeneous tenants with the same satisfaction function and maximum bandwidth demand, and (ii) heterogeneous tenants with different satisfaction functions competing to allocate their individual share of the bandwidth in link l. For homogeneous tenants, we consider strictly concave satisfaction functions of tenant k, $\mathcal{U}_k = \ln(1 + b_k)$, where b_k is measured in Gbps. For heterogeneous tenants, the satisfaction function of tenant k is considered as $\mathcal{U}_k = \ln(1 + (1 + (\frac{k}{K}))b_k)$. In general, the cloud provider does not have the information of $\mathcal{U}_k$ of the tenants. In this case, it uses binary search algorithm and the best response strategy to find the cost factors β and α that achieve the desired congestion degree C_d.

In our experiments on Amazon EC2 [21], there are $K = 8$ heterogeneous tenants and 16 *t2.micro* VM instances (1 CPU core and 1GB memory). We used Python socket programming to create a TCP communication between each of the 8 VM pairs of each tenant on a link l with $B_l^{\max} = 40$ Mbps. We consider a scenario in which each tenant tries to send data from one VM to the other VM. We set the number of updates in the asynchronous-based iterative bandwidth update process to 1000, which is large enough to converge to the optimal bandwidth schedule. In order to emulate the cloud provider, we used another VM on Amazon EC2. Unless otherwise indicated, the settings in the real experiments are the same as those in the simulation.

B. Experimental Results

In the simulation and real implementation, we measured the social welfare, average bandwidth payment, and total satisfaction of the tenants. We varied the desired congestion degree C_d from 0.1 to 1 with step increase of 0.1. Then, we determined α and β as explained previously to achieve



(a) Social welfare (b) Total bandwidth payments (c) Total satisfaction (d) Congestion degree

Fig. 6: The experimental results of heterogeneous tenants in real implementation.

this specified congestion degree. After the optimal bandwidth schedule is determined, we calculated the social welfare, the total bandwidth payment and total satisfaction of the tenants based on (4), (17).

Figures 4(a) and 5(a) show the social welfare of homogeneous and heterogeneous tenants versus the desired congestion degrees in the simulation, respectively. Figure 6(a) shows the social welfare of heterogeneous tenants versus the desired congestion degree in the real implementation. We see that the social welfare is higher with nonlinear pricing model than with linear pricing model. It demonstrates that nonlinear pricing policy provides better social welfare, i.e., higher QoS, to the tenants. Through nonlinear pricing model, the tenants are more likely to be satisfied with the cloud provider, and hence the cloud provider can further attract more tenants to use its services. Furthermore, we see that using the nonlinear pricing policy, the social welfare increases as the number of tenants increases for both homogenous and heterogenous cases. However, using the linear pricing policy, the social welfare does not necessarily increase as the number of tenants increases for homogeneous tenants. It indicates that with the nonlinear pricing policy, more tenants lead to higher social welfare.

Figures 4(b) and 5(b) show the total bandwidth payment of homogeneous and heterogeneous tenants versus the desired congestion degree. Figure 6(b) shows the total bandwidth payment of heterogeneous tenants versus the congestion degree in the real implementation. We see that the average bandwidth payment with the linear congestion cost function is slightly higher than that with the nonlinear congestion cost function. It shows that using the nonlinear pricing policy, the cloud provider assigns less bandwidth cost for the tenants to achieve a desired congestion degree in link l. Paying less amount of the bandwidth payment by the tenants makes them more satisfied with the service of the cloud provider. Consequently, it encourages the tenants to choose the cloud provider for their services.

Figures 4(c) and 5(c) show the total satisfaction of the homogeneous tenants and heterogeneous tenants in the simulation, respectively. Figure 6(c) shows the total satisfaction of the heterogeneous tenants in the real implementation. We see that for both homogeneous and heterogeneous tenants, the total satisfaction is equal for both linear and nonlinear congestion cost functions, since at a fixed congestion degree, the term $\mathcal{V}(\sum_k b_k)$ in (4) is fixed at the same value for both linear and nonlinear pricing policies. Thus, the optimization problem in (4) is equivalent with maximizing the first term in (4) that is the total satisfaction of the tenants. In addition, we see that at a fixed desired congestion degree, the total satisfaction increases as the number of tenants increases due to the concavity of the satisfaction function. This shows that the cloud provider prefers to have more tenants using its service.

We then measured the convergence speed for the nonlinear and linear pricing policies when the desired congestion degree was set to 90%. We define congestion error as $|\frac{C_d^m - C_d}{C_d}|$, where C_d^m is the congestion degree for the bandwidth allocations at m^{th} update, and C_d is the desired congestion degree. Figures 4(d) and 5(d) plot the congestion degree for the bandwidth allocations at m^{th} update versus the number of updates (i.e., m) to achieve the convergence error of 1%. Figures 4(d) and 5(d) show the congestion degree of link l as the number of updates increases in the simulation. These figures are obtained by taking average of 1000 times of experiment runs. Figure 6(d) shows the congestion degree as the number of updates increases for heterogeneous tenants in the real implementation. As it is seen, the congestion degree converges faster using the nonlinear pricing policy, and it needs fewer updates to converge than the linear pricing policy for all different number of tenants. As we discussed in Section III, using nonlinear pricing policy, the bandwidth payment cost increases as the congestion degree increases. Thus, for the less congested link, the tenants pay less for a specific amount of bandwidth. Therefore, the tenants allocate more bandwidth when the link is less congested. In contrast, the bandwidth payment does not change with the congestion degree when we use the linear pricing policy. As a result, the congestion degree converges to the desired congestion degree faster than using the linear pricing policy as shown in figures 4(d), 5(d) and 6(d). In order to show the real time duration for the iterative update process, we measured the total time it took to complete the update process. In this test, we set the desired congestion degree to 0.9 and varied the number of updates from 20 to 10000. Figure 7 shows the update process duration as the number of updates increases. We see that it takes around one second to complete 1000 updates (a sufficient number of updates as shown in Figure 6(d)), which is a short time to converge to the optimal solution. The result means that for each short-time period (e.g., 5 minutes), nonlinear pricing policy spends around one second to determine the bandwidth schedule for this period. As it is seen, the update process duration increases logarithmically as the number of updates increases. Further numerical experiments also show that the number of updates increases linearly as the number of tenants increases.

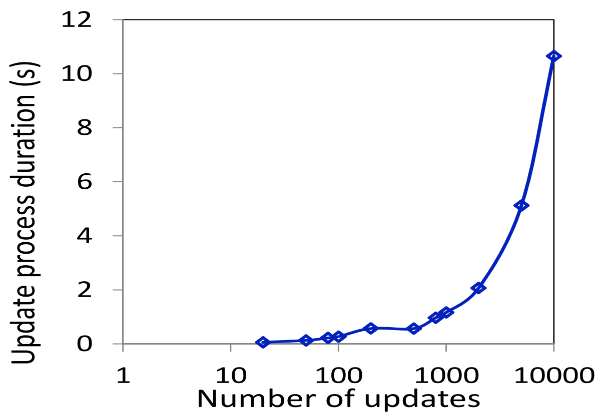


Fig. 7: Update process duration of a tenant in real implementation.

IX. Conclusion and Future Work

In this paper we proposed nonlinear bandwidth pricing for congestion control in a communication link in the cloud networks. We defined the social welfare of the tenants as the total satisfaction functions of the tenants minus the congestion cost over the link. We formulated the utility optimization of the tenants as a strategic game, and we showed different properties of the Nash equilibrium of the designated game. We showed that the Nash equilibrium exists, and it is uniquely equal to the social optimum bandwidth schedule that maximizes the social welfare of the tenants.

As the tenants are not obliged to truthfully reveal their private information such as maximum required bandwidth and satisfaction function, we implemented a decentralized asynchronous-based best response strategy to find the Nash equilibrium bandwidth schedule. We showed that the best response strategies of the tenants converge to the Nash equilibrium bandwidth schedule under reasonable mild assumptions such as individually selfish utility maximizing tenants. We also derived the total bandwidth payments of the tenants, and discussed that the total bandwidth payments is maximized at the fully congested link. We verified our results with numerical simulations for different number of the tenants.

In this paper, we considered different tenants sharing one bandwidth limited link in the cloud network. We used the overload cost function to satisfy the bandwidth maximum capacity constraint. In our future work, we will consider the scenario that the tenants share multiple links in the network. We will also use adaptive control methods to satisfy the bandwidth capacity constraint.

X. Acknowledgement

This research was supported in part by U.S. NSF grants NSF-1404981, IIS-1354123, CNS-1254006, IBM Faculty Award 5501145 and Microsoft Research Faculty Fellowship 8300751.

References

[1] M. Armburst and et. al, "A view of cloud computing," *Commun. ACM*, vol. 53, no. 4, pp. 50–58, Apr. 2010.
[2] D. Kossmann, T. Kraska, and S. Loesing, "An evaluation of alternative architectures for transaction processing in the cloud," in *Proc. of SIGMOD*, 2010, pp. 579–590.
[3] D. Niu, C. Feng, and B. Li, "Pricing cloud bandwidth reservations under demand uncertainty," in *Proc. of SIGMETRICS*, 2012.
[4] H. Shen and Z. Li, "New bandwidth sharing and pricing policies to achieve a win-win situation for cloud provider and tenants," in *Proc. IEEE INFOCOM*, Toronto, CA, Apr. 2014, pp. 835–843.
[5] ——, "New bandwidth sharing and pricing policies to achieve a win-win situation for cloud provider and tenants," *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, vol. 27, no. 9, pp. 2682–2697, 2016.
[6] J. Guo, F. Liu, D. Zeng, J. C. Lui, and H. Jin, "A cooperative game based allocation for sharing data center networks," in *Proc. IEEE INFOCOM*, Turin, Italy, April 2013, pp. 2139–2147.
[7] H. Ballani, P. Costa, T. Karagiannis, and A. Rowstron, "Towards predictable datacenter networks," in *Proc. SIGCOMM*, New York, NY, Aug. 2011, pp. 242–253.
[8] H. Shen and T. Basar, "Incentive-based pricing for network games with complete and incomplete information," *Advances in Dynamic Game Theory*, vol. 9, pp. 431–458, 2007.
[9] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *J. Operational Research Soc.*, vol. 49, no. 3, pp. 237–252, March 1998.
[10] S. H. Low and D. E. Lapsley, "Optimization flow control, i: Basic algorithm and convergence," *IEEE/ACM Tran. Netw.*, vol. 7, no. 6, pp. 861–874, December 1999.
[11] A. Ghavami, Z. Li, and H. Shen, "On-demand bandwidth pricing for congestion control in core switches in cloud networks," in *IEEE Proc. CLOUD*, June 2016.
[12] L. Popa, G. Kumar, M. Chowdhury, A. Krishnamurthy, S. Ratnasamy, and I. Stoica, "Faircloud: Sharing the network in cloud computing," in *Proc. ACM SIGCOMM*, New York, NY, 2012, pp. 187–198.
[13] "What we talk about when we talk about cloud network performance," in *Proc. ACM SIGCOMM*, October 2012, pp. 44–48.
[14] "a flexible model for resource management in virtual private networks," in *Proc. of ACM SIGCOMM*, 1999.
[15] H. Jin, X. Wang, S. Wu, S. Di, and X. Shi, "Towards optimized fine-grained pricing of iaas cloud platform," *IEEE Transactions on cloud Computing*, 2014.
[16] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA: Athena Scientific, 1999.
[17] D. Fundenberg and J. Tirole, *Game Theory*, 1st ed. Cambridge, MA: The MIT Press, 1991.
[18] M. J. Osborne and A. Rubinstein, *A Course in Game Theory*, 1st ed. Cambridge, MA: The MIT Press, 1994.
[19] D. Xie, N. Ding, Y. C. Hu, and R. Kompella, "The only constant is change: Incorporating time-varying network reservations in data centers," in *Proc. of SIGCOMM*, 2012.
[20] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, 1st ed. Athena Scientific, 1989.
[21] "Amazon elastic compute cloud," http://aws.amazon.com/ec2, [Accessed in June 2015].