Non-convexity Issues for Internet Rate Control with Multi-class Services: Stability and Optimality

Jang-Won Lee, Ravi R. Mazumdar, and Ness B. Shroff
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907, USA
{lee46, mazum, shroff}@ecn.purdue.edu

***Abstract*— In this paper, we investigate the problem of distributively allocating transmission rates to users on the Internet. We allow users to have concave as well as sigmoidal utility functions that are natural in the context of various applications. In the literature, for simplicity, most works have dealt only with the concave case. However, we show that when applying rate control algorithms developed for concave utility functions in a more realistic setting (with both concave and sigmoidal types of utility functions), they could lead to instability and high network congestion. We show that a pricing based mechanism that solves the dual formulation can be developed based on the theory of subdifferentials with the property that the prices "self-regulate" the users to access the resource based on the net utility. We discuss convergence issues and show that an algorithm can be developed that is *efficient* in the sense of achieving the global optimum when there are many users.**

I. Introduction

There has been a lot of interest in the area of Internet rate control. Most Internet services are elastic to some degree, i.e., the sources can adjust their transmission rates in response to congestion levels within the network. Hence, by appropriately exploiting the elasticity through rate control, one can maintain high network efficiency while at the same time alleviating network congestion. To that end, it is necessary to have an appropriate model to characterize the elasticity of the service. This is typically done using the well known concept of a utility function that represents the level of user satisfaction or Quality of Service (QoS) at the allocated rate.

We can classify services in the Internet into two classes based on the shape of the utility function. One corresponds to traditional data services, such as file transfer and email. These services can adjust their transmission rates gradually, resulting in graceful degradation of the QoS in the presence of network congestion. The elasticity of these services can be modeled by concave utility functions [1]. The other corresponds to real-time services, such as streaming video and audio services. These services are less elastic than data services. In response to network congestion, they can decrease their transmission rates up to a certain level with a corresponding graceful degradation in the QoS. However, decreasing the transmission rate below a certain threshold results in a significant drop in the QoS (e.g., below a certain bit rate, the quality of audio communication falls dramatically). The elasticity of these services can be modeled by using sigmoidal-like utility functions [1]. We call an increasing function $f(x)$ a *sigmoidal-like function*, if it has one inflection point x_o, and $\frac{d^2f(x)}{dx^2} > 0$, for $x < x_o$ and $\frac{d^2f(x)}{dx^2} < 0$, for $x > x_o$, as shown in Fig. 1.

There have been a number of papers that have studied utility based rate control problems by exploiting the elasticity of services, e.g., [2], [3], [4], [5], [6], [7], [8], [9]. Most of these works use a utility and pricing framework that attempts to obtain the optimal rate allocation that maximizes the total system utility using the price as a control signal. In this framework, the network announces the price that measures the congestion level of the network to the users. Based on this price, each user adjusts its transmission rates in an attempt to maximize its net utility, which is defined by

$$U(x) - \lambda x,$$

where U is a utility function, λ is price for unit rate, and x is the amount of rate allocation.

In [2], the author shows that the problem can be decomposed into the user problem and the network problem. Based on the decomposed problem, the authors in [3] propose a distributed algorithm with a penalty function method that converges to the global optimal rate allocation. They also show that the

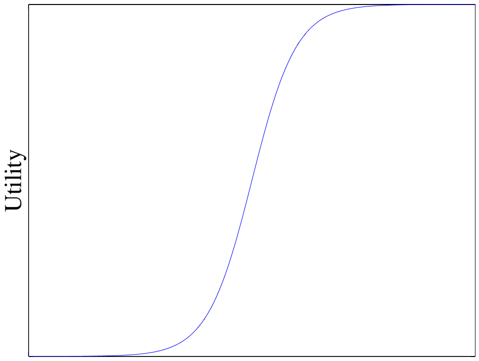


Fig. 1. A sigmoidal-like function.

This research has been supported in part by NSF grants ANI-0073359, ANI-9805441, and ANI-0207728.

rates per unit charge are proportionally fair. In [4], the authors consider the same problem as in [3] but solve it using the dual. Since the primal is a convex programming problem, there is no duality gap between it and its dual. Hence, by solving the dual problem, the optimal primal solution (the optimal rate allocation) can be obtained. In this paper, the authors use a gradient projection algorithm for the dual and show that the algorithm converges to the optimal solution. Their algorithm is implemented using Random Exponential Marking (REM) in a following paper [5]. In [6], the authors develop a Nash bargaining solution that is proportionally fair and Pareto-optimal. The authors solve the dual problem using a gradient projection algorithm and implement the algorithm with resource management packets in the Asynchronous Transfer Mode (ATM) network context. In [7], the authors use a similar approach as in [3]. However, they consider the problem with random loss in the network and implement the algorithm using Explicit Congestion Notification (ECN) marking. In [8], a window based algorithm is proposed by generalizing the works in [3] and [10]. In [9], the authors propose a subgradient based algorithm using the number of congested links on the user's path as an indicator of network congestion.

The common feature in the afore-mentioned works is that they all require the utility functions to be concave, resulting in a convex programming problem. However, as mentioned before, concave utility functions are appropriate only for modeling traditional data services, and do not capture the characteristics of services such as audio and video that are becoming increasingly popular on the Internet. Hence, for the efficient allocation of transmission rates among services with diverse characteristics, a rate control algorithm must be able to efficiently handle real-time services with sigmoidal-like utility functions as well as data services with concave utility functions.

In this paper, we will study this problem by considering a situation similar to the current Internet. In the Internet, there is no central authority in the system that performs admission control or rate control and each user behaves in a selfish manner. Thus, a rate control algorithm must be implemented in a distributed manner taking into account the selfish behavior of users. In the papers mentioned earlier, it has been shown that if all users have concave utility functions, efficient distributed rate allocations can be obtained using an appropriate congestion indicator in the network in spite of the selfish behavior of users. However, as we will show later, if such algorithms developed for concave functions are now applied to non-concave functions, the system can become unstable and could cause excessive congestion in the network.

A seemingly logical approach to deal with the issue of non-convexity is to simply approximate a sigmoidal-like utility function with a concave function and use one of algorithms developed for concave utility functions. However, this approach could result in a highly inefficient solution. For example, suppose that a system has a single bottleneck link with capacity 10 Mbps and 11 users. Further, suppose that each user has the same utility function $U(x)$ that is a step function described below:

$$U(x) \quad = \quad \begin{cases} 0, & \text{if } x < 1 \text{ Mbps} \\ 1, & \text{if } x \geq 1 \text{ Mbps} \end{cases} .$$

Note that the step function is an extreme case of a sigmoidal-like function. Let us approximate $U(x)$ with a concave function, $U'(x)$. Then, we can apply an algorithm for concave utility functions that has been proposed in the literature to maximize the total system utility. In this case, since all users have the same utility functions, at the global optimal solution, each user is allocated the same amount of rate, $x^* = \frac{10}{11}$ Mbps, which provides $U(x^*) = 0$. Hence, with this approach, we achieve zero total system utility for the original utility function. However, by allocating 1 Mbps to 10 users and zero to one user, we can achieve a total system utility of 10 units. Even though this example considers an extreme case, it emphasizes that to efficiently accommodate diverse services in the Internet, it is necessary to develop a rate allocation algorithm that takes into account the properties of both concave and sigmoidal-like utility functions. In this paper, we will study this problem and focus on issues of convergence and efficiency.

The rest of the paper is organized as follows. In Section II, we describe the system model and present the problem that is being considered in this paper. We propose and study the rate control algorithm in Section III. For the sake of brevity, proofs are omitted. Interested readers are referred to our technical report [11]. We provide numerical results for the proposed algorithm in Section IV and conclude in Section V.

II. System description and basic problem

We consider a system that has a single bottleneck link with capacity C, as shown in Fig. 2. There are N users that use the bottleneck link. Each user i has a utility function U_i and maximum transmission rate M_i ($0 < M_i < \infty$). We assume that U_i has the following properties.

Properties of the utility function:

(U1) U_i is an increasing function of x_i the *allocated rate* for user i.

(U2) U_i is a sigmoidal-like or strictly concave function.

(U3) U_i is continuous and differentiable.

In the following, if U_i is a sigmoidal-like function, we let x_i^o be its inflection point. Otherwise, i.e., U_i is a concave function, we let $x_i^o = 0$.

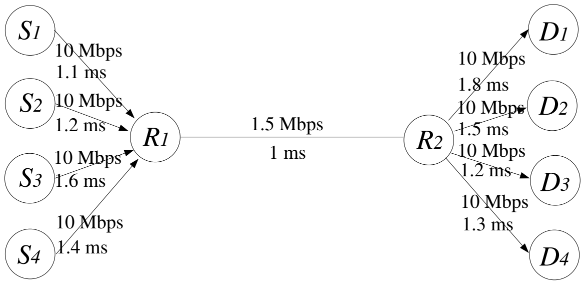


Fig. 2. A system with a single bottleneck link.

Our objective is to obtain a transmission rate allocation for each user that maximizes the total system utility. This is formulated as:

$$
\begin{aligned}
\text{(A)} \qquad & \max \sum_{i=1}^{N} U_i(x_i) \\
& \text{subject to} \quad \sum_{i=1}^{N} x_i \leq C \\
& \qquad\qquad\quad 0 \leq x_i \leq M_i, \forall\, i,
\end{aligned}
$$

where x_i is the allocated data rate for user i. To avoid trivialities, we will assume throughout the paper that $\sum_{i=1}^{N} M_i > C$. Note that since we allow non-concave utility functions, problem (A) is a non-convex programming problem, which, usually, is more difficult to solve than a convex programming problem. In [12], a similar problem to (A) was studied in the context of the power allocation in wireless environment. However, the algorithm in [12] requires a central controller, such as a base-station in cellular systems, which is clearly not applicable to decentralized networks, such as the Internet. In this paper, we will take a different approach to solve problem (A), resulting in a decentralized solution.

III. Rate control

In this section, we develop a distributed rate control algorithm for problem (A). We will use the theory of subdifferentials in this paper. For background, we first provide definitions and properties of subdifferentials. We refer readers to [13], [14], [15] for details.

Definition 1: A vector $d \in R^n$ is a subgradient of a convex function $f : R^n \to R$ at $x \in R^n$, if

$$f(z) \geq f(x) + (z - x)^T d, \text{ for all } z \in R^n.$$

Definition 2: The set of all subgradients of a convex function f at $x \in R^n$ is called the subdifferential of f at x and denoted by $\partial f(x)$.

Properties of the subgradient:

(S1) A function $f(x)$ is differentiable at x, if and only if it has a unique subgradient at x. In this case, the subgradient is equal to the gradient of f at x.

(S2) $x \in X \subset R^n$ minimizes a convex function f over a convex set X, if and only if there exists a subgradient d such that $d^T(z-x) \geq 0$, for all $z \in X$, where d^T is a transpose of a vector d.

(S3) If x is an interior point of X, then (S2) implies that x minimizes a convex function f over a convex set X, if and only if $0 \in \partial f(x)$.

(S4) $f'(x;y) = \max_{d \in \partial f(x)} y^T d$, for all $y \in R^n$, where $f'(x;y)$ is a directional derivative of f at x in the direction y and defined by

$$f'(x;y) = \lim_{h \downarrow 0} \frac{f(x+hy) - f(x)}{h}.$$

A. Dual problem

As mentioned before, problem (A) is a non-convex programming problem, which is difficult to solve. We will consider its dual since the dual has some advantages over the primal problem. For example:

- The dual problem is a convex programming problem and is thus easier to solve.
- The separable property of the dual problem makes it easy to implement the algorithm in a distributed fashion.
- From a networking perspective, the dual problem will usually have a smaller dimension and simpler constraints than the primal problem. This will reduce the complexity of the algorithm. In our case, the primal problem has a dimension of N and the dual problem has a dimension of L, where N is the number of users in the network and L is the number of links in the network. In general, we have $L \ll N$[1].

However, since the primal is not a convex programming problem (e.g., if some of the utility functions are sigmoidal), there could be a duality gap between the primal and its dual. Hence, by solving the dual, we may not obtain the optimal primal solution. This is one of the difficulties that we will overcome in this work, especially in the context of many users.

We now define a Lagrangian function associated with problem (A) as:

$$L(\bar{x}, \lambda) \quad = \quad \sum_{i=1}^{N} U_i(x_i) + \lambda(C - \sum_{i=1}^{N} x_i), \tag{1}$$

where $\bar{x} = (x_1, x_2, \cdots, x_N)^T$. Then, the dual of problem (A) can be defined as:

$$
\begin{aligned}
\text{(B)} \qquad & \min Q(\lambda) \\
& \text{subject to} \quad \lambda \geq 0,
\end{aligned}
$$

where

$$Q(\lambda) \quad = \quad \max_{\bar{0} \leq \bar{x} \leq \bar{M}} L(\bar{x}, \lambda), \tag{2}$$

$\bar{M} = (M_1, M_2, \cdots, M_N)^T$, $\bar{0} = (0, 0, \cdots, 0)^T$, and the inequality between the two vectors is a component-wise inequality (i.e., $\bar{x} \leq \bar{M}$ implies $x_i \leq M_i$, $\forall i$). It can easily be shown that $Q(\lambda)$ is a convex function of λ [15]. However, as we will show later, $Q(\lambda)$ may not be everywhere differentiable. Hence, even though $Q(\lambda)$ is a convex function, we cannot use a simple gradient based algorithm to find a minimizer as in [4], [6], since $Q(\lambda)$ does not have a gradient at the point where it is not differentiable.

To solve problem (B), we will first study the properties of $Q(\lambda)$ by using the theory of the subdifferentials. We now characterize the subdifferentials of $Q(\lambda)$. Note that $L(\bar{x}, \lambda)$ in (1) is separable in $\bar{x}$. Thus, $\bar{x}(\lambda)$ solves (2) if and only if it solves the following:

$$x_i(\lambda) \quad = \quad \arg \max_{0 \leq x \leq M_i} \{U_i(x) - \lambda x\}, \forall\, i. \tag{3}$$

[1] Since, in this paper, we focus on a single bottleneck link, $L = 1$

The properties of $x_i(\lambda)$ were studied in [12]. First, we define λ_i^{max} for user i as:

$$\lambda_i^{max} = \min\{\lambda \geq 0 \mid \max_{0\leq x\leq M_i}\{U_i(x) - \lambda x\} = 0\}. \qquad (4)$$

We can calculate it by the following equation:

$$\lambda_i^{max} = \begin{cases} \frac{dU_i(x)}{dx}|_{x=0}, & \text{if } x_i^o = 0, \\ \frac{dU_i(x)}{dx}|_{x=x'}, & \text{if } 0 < x_i^o < M_i \text{ and } x' \text{ exists}, \\ \frac{U_i(M_i)}{M_i}, & \text{otherwise}, \end{cases}$$

where x' is a solution of the following equation:

$$U_i(x) - x\frac{dU_i(x)}{dx} = 0,\ x_i^o \leq x \leq M_i.$$

Also, define λ_i^{min} for user i as:

$$\lambda_i^{min} = \max\{\lambda \geq 0 | x_i(\lambda) = M_i\}.$$

Obviously, $\lambda_i^{max} > 0$ and $\lambda_i^{max} \geq \lambda_i^{min}$. Then, $x_i(\lambda)$ has the following properties.

Properties of $x_i(\lambda)$:

(R1) If U_i is a sigmoidal-like function (i.e., $x_i^o > 0$), $x_i(\lambda)$ has two values (zero and positive) and is discontinuous at λ_i^{max}. Otherwise, $x_i(\lambda)$ has a unique value and is continuous.

(R2) $x_i(\lambda)$ is positive and a decreasing function of λ, for $\lambda_i^{min} \leq \lambda < \lambda_i^{max}$.

(R3) $x_i(\lambda)$ is zero, for $\lambda > \lambda_i^{max}$.

(R4) $x_i(\lambda)$ is M_i, for$\lambda \leq \lambda_i^{min}$.

(R5) $U_i(x_i(\lambda_i^{max}))$ is achieved at the concave region of U_i.

Note that, if U_i is a concave function, $x_i(\lambda)$ is a continuous and non-increasing function. However, if U_i is a sigmoidal-like function, $x_i(\lambda)$ is not only discontinuous but also has two values at λ_i^{max}. In the sequel, $x_i(\lambda)$ always implies a positive value, if (3) has two solutions.

Since the Lagrangian function, $L(\bar{x}, \cdot)$, is differentiable for all $\bar{0} \leq \bar{x} \leq \bar{M}$, and $\frac{\partial L(\cdot,\lambda)}{\partial\lambda}$ is continuous on $\bar{0} \leq \bar{x} \leq \bar{M}$ for each $\bar{x}$, by Danskin's Theorem [15] the subdifferential of $Q(\lambda)$, $\partial Q(\lambda)$, is obtained as:

$$\partial Q(\lambda) \quad = \quad \text{conv}(\{\frac{\partial L(\bar{x},\lambda)}{\partial\lambda} \mid \bar{x} \in \bar{x}(\lambda)\}), \qquad (5)$$

where $\bar{x}(\lambda)$ is a set of solutions of (3) at λ, and $\text{conv}(G)$ is a convex hull of a set G. Hence, by using the properties of $x_i(\lambda)$, the subdifferential of Q at λ is obtained by

$$\partial Q(\lambda) = \begin{cases} \{d \mid C - \sum_{j\in H_i\cup S_i} x_j(\lambda_i^{max}) & \text{if } x_i^o > 0 \text{ and} \\ \leq d \leq C - \sum_{j\in H_i} x_j(\lambda_i^{max})\}, & \lambda = \lambda_i^{max}, \\ \{C - \sum_{j=1}^{N} x_j(\lambda)\}, & \text{otherwise}, \end{cases} \qquad (6)$$

where we divide the set of users into three subsets of users corresponding to each user i as

$$H_i \quad = \quad \{j \mid \lambda_j^{max} > \lambda_i^{max},\ 1 \leq j \leq N\}, \qquad (7)$$

$$S_i \quad = \quad \{j \mid \lambda_j^{max} = \lambda_i^{max},\ 1 \leq j \leq N\}, \text{ and} \qquad (8)$$

$$L_i \quad = \quad \{j \mid \lambda_j^{max} < \lambda_i^{max},\ 1 \leq j \leq N\}. \qquad (9)$$

We now solve the dual problem (B). The next proposition shows us that problem (B) has a unique solution.

Proposition 1: The dual problem (B) has a unique optimal solution $\lambda^o > 0$.

However, as shown in (6), if $x_i^o > 0$ for U_i (i.e., U_i is a sigmoidal-like function), then the subgradient of $Q(\lambda)$ at λ_i^{max} is not unique, while, for concave utility functions, the subgradient of $Q(\lambda)$ is unique. This implies that, by property (S1), if U_i is a sigmoidal-like function, $Q(\lambda)$ is not differentiable at $\lambda = \lambda_i^{max}$ and, otherwise, $Q(\lambda)$ is differentiable. Hence, there may not exist a gradient of $Q(\lambda)$ for all $\lambda \geq 0$. Thus, we cannot use a gradient based method to solve problem (B) and we will consider a subgradient projection method, which is formulated using an iterative algorithm such as:

$$\lambda^{(n+1)} \quad = \quad [\lambda^{(n)} - \alpha^{(n)}(C - \sum_{i=1}^{N} x_i(\lambda^{(n)}))]^+, \qquad (10)$$

where $x_i(\lambda^{(n)})$ is a solution of (3) at $\lambda = \lambda^{(n)}$ and $[a]^+ = \max\{a, 0\}$. By (6), $C - \sum_{i=1}^{N} x_i(\lambda^{(n)})$ is a subgradient of $Q(\lambda)$ at $\lambda = \lambda^{(n)}$. To make (10) converge to λ^o, the optimal solution of problem (B), we must have an appropriate sequence of $\alpha^{(n)}$. In gradient based algorithms in [4], [6], there exists a constant step size, $\alpha^{(n)} = \alpha$, such that $\lambda^{(n)}$ converges to λ^o. However, in the subgradient based algorithm, we cannot guarantee convergence of $\lambda^{(n)}$ using a constant step size, since the subgradient, $C - \sum_{i=1}^{N} x_i(\lambda^{(n)})$, that we use in (10), may not be zero at λ^o. Hence, we will consider the following sequence:

$$\alpha^{(n)} \to 0,\ \text{as } n \to \infty \quad \text{and} \quad \sum_{n=1}^{\infty} \alpha^{(n)} = \infty. \qquad (11)$$

We can then show (as given by the next Proposition) that $\lambda^{(n)}$ in (10) converges to the optimal solution λ^o of the dual problem (B), with the sequence that satisfies conditions in (11).

Proposition 2: The series of $\lambda^{(n)}$ that are generated by (3) and (10) with $\alpha^{(n)}$ in (11) converge to λ^o, the optimal solution of problem (B).

B. Distributed algorithm for the dual problem

In the previous subsection, we have established that the solution of (3) and (10) with coefficients satisfying (11) converges to the dual optimal solution. This algorithm can be implemented in a distributed way. At iteration n, user i transmits its data at a rate determined by solving (3) with $\lambda = \lambda^{(n)}$. In this case, we can interpret $\lambda^{(n)}$ as the price per unit rate at iteration n. With this interpretation, by solving

(3), user i tries to maximize its net utility without considering other users. This is a natural property of selfishness (non-cooperative property) of the user in a public environment, such as the Internet. Also, we can interpret λ_i^{max} as the maximum willingness to pay per unit rate of user i, since the price per unit rate λ is higher than λ_i^{max}, $x_i(\lambda)$ will be zero by property (R3) (i.e., user i does not transmit its data.). Note that the utility and the net utility must be calculated with the allocated rate. However, the user cannot know its allocated rate in advance before it transmits data. Thus, the user maximizes its net utility with the transmission rate assuming that the allocated rate is same as the transmission rate. Based on the aggregate transmission rate of users, a node updates the price per unit rate of the next iteration by solving (10). This implies that a node tries to obtain the optimal price per unit rate that solves the dual problem by adjusting it based on the congestion level. Also, the node tries to maximize the utilization of its capacity without causing congestion by equating the aggregate transmission rate of users with its capacity.

C. Properties of the primal solution

Thus far, we have considered the dual of problem (A) and proposed an algorithm that converges to an optimal solution λ^o of the dual problem. When there is no duality gap between the primal problem (A) and its dual (B), this results in an optimal solution of the primal, since $\bar{x}(\lambda^o)$ is an optimal solution of the primal problem (A). However, when some of the utilities are non-concave, the primal problem (A) is not a convex programming problem. In which case there could exist a duality gap between the primal and its dual, i.e., the solution of the dual problem (B) need not be the optimal solution of the primal problem (A). In this paper, we are more interested in the rate allocation (the primal solution) than the price (the dual solution). Thus, it is important to study how "good" a primal solution can be obtained by solving its dual. To that end, we next study the properties of the primal solution corresponding to its dual optimal solution.

Proposition 3: Suppose that λ^o is an optimal solution of the dual problem (B). Then, if (3) has a unique solution at λ^o for all i and $\sum_{i=1}^N x_i(\lambda^o) = C$, then, $\bar{x}(\lambda^{(n)})$ converges to $\bar{x}(\lambda^o)$. Moreover, it is a global optimal rate allocation. Otherwise, $\bar{x}(\lambda^{(n)})$ may not converge, even though $\lambda^{(n)}$ converges to λ^o.

If the condition in Proposition 3 is satisfied, by solving the dual problem (B), we can obtain an optimal solution of problem (A). However, if the condition in Proposition 3 is not satisfied, we cannot obtain a primal optimal solution by solving its dual. This happens when there exists a user k^o such that $x_{k^o}^o > 0$ and one of the following conditions are satisfied:

$$\begin{aligned}
&\text{(i)} && \sum_{i \in H_{k^o}} x_i(\lambda_{k^o}^{max}) < C - \epsilon_1 \text{ and} \\
& && \sum_{i \in H_{k^o} \cup S_{k^o}} x_i(\lambda_{k^o}^{max}) > C + \epsilon_2, \\
&\text{(ii)} && \sum_{i \in H_{k^o}} x_i(\lambda_{k^o}^{max}) \le C \text{ and} \\
& && \sum_{i \in H_{k^o} \cup S_{k^o}} x_i(\lambda_{k^o}^{max}) > C + \epsilon_3, \text{ or} \qquad (12) \\
&\text{(iii)} && \sum_{i \in H_{k^o}} x_i(\lambda_{k^o}^{max}) < C - \epsilon_4 \text{ and} \\
& && \sum_{i \in H_{k^o} \cup S_{k^o}} x_i(\lambda_{k^o}^{max}) \ge C,
\end{aligned}$$

where ϵ_1, ϵ_2, ϵ_3, and ϵ_4 are some positive constants. In this case, since $0 \in \partial Q(\lambda_{k^o})$, λ_{k^o} is a dual optimal solution and $\lambda^{(n)}$ converges to it. However, when $\lambda^{(n)}$ converges to λ_{k^o}, the rate allocation oscillates between the feasible solution and the infeasible solution, and does not converge. Since the rate allocation can be infeasible, (i.e., the aggregate transmission rate of users can exceed the capacity of the node), congestion may occur in the node. Note that this situation happens because of the discontinuity of $x_i(\lambda)$ when U_i is a sigmoidal-like function. Thus, if there exist users having sigmoidal-like utility functions, the rate allocation resulting from solving the dual problem, such as the algorithms in [4], [6] (that converges to an efficient rate allocation with concave utility functions), may cause congestion without convergence.

To resolve this situation, we will impose a "self-regulating" property on the users. In the next subsection, we will study the "self-regulating" property and show that using the "self-regulating" property, the algorithm converges to the feasible primal solution that is an asymptotically optimal rate allocation.

D. "Self-regulating" property

To study the "self-regulating" property, we assume that the condition in Proposition 3 is not satisfied in this subsection. Thus, there exist user k^o and subsets of users, H_{k^o}, S_{k^o}, and L_{k^o} that satisfy one of conditions in (12).

We first define what we mean by the "self-regulating" property and make additional assumptions for the convergence of the algorithm having the "self-regulating" property.

Definition 3: The property of a user that it does not transmit data even though the price is less than its maximum willingness to pay, if it realizes that it will receive non-positive net utility is called the "self-regulating" property.

Note that, with the "self-regulating" property, users continue to be selfish, i.e., they still preserve the non-cooperative property.

Assumptions:

- (A1) Each user is "self-regulating", i.e., it satisfies the "self-regulating" policy.
- (A2) Each user i has a threshold of tolerance th_i such that if it does not transmit data because the price is higher than its maximum willingness to pay or if it receives negative net utility by transmitting data for th_i iterations consecutively (i.e., it receives non-positive net utility for th_i iterations consecutively), it stops transmitting data.

(A3) The node allocates a rate to each user as

$$x_i'(\lambda) \;=\; \begin{cases} x_i(\lambda), & \text{if } \sum_{j=1}^{N} x_j(\lambda) \le C, \\ f_i(\bar{x}(\lambda)), & \text{if } \sum_{j=1}^{N} x_j(\lambda) > C, \end{cases}$$

where $x_i(\lambda)$ is the transmission rate of user i at price λ and f_i is a continuous function of $\bar{x}$ that satisfies the following conditions:

$$f_i(\bar{x}) < x_i \quad \text{and} \quad \sum_{i=1}^{N} f_i(\bar{x}) = C.$$

A good candidate for function f_i is

$$f_i(\bar{x}) \;=\; \frac{x_i}{\sum_{j=1}^{N} x_j} C,$$

which can be achieved by the First Come First Service (FCFS) policy.

We first define the net utility for user i at price λ and received rate r as

$$NU_i(\lambda, r) \;=\; U_i(r) - \lambda r$$

and the maximum net utility of user i at the price λ as

$$NU_i^{max}(\lambda) \;=\; \max_{0 \le r \le M_i} \{U_i(r) - \lambda r\}.$$

We now assume that the algorithm in (3) and (10) is at the mth iteration such that $\lambda_{L_{k^o}}^{max} < \lambda^{(n)} < \lambda_{H_{k^o}}^{max}$ for all $n \ge m$, where $\lambda_{L_{k^o}}^{max} = \max_{i \in L_{k^o}} \{\lambda_i^{max}\}$ and $\lambda_{H_{k^o}}^{max} = \min_{i \in H_{k^o}} \{\lambda_i^{max}\}$. Since $\lambda^{(n)}$ converges to $\lambda_{k^o}^{max}$, the dual optimal solution and $\lambda_{L_{k^o}}^{max} < \lambda_{k^o}^{max} < \lambda_{H_{k^o}}^{max}$, there exists such an m that satisfies the above condition. In this case, users in set L_{k^o} do not transmit data, since the price is higher than their maximum willingness to pay. We can divide the situations into two cases. First, suppose that $\lambda^{(n)} > \lambda_{k^o}^{max}$. Then, users in set S_{k^o} do not transmit data, since the price is higher than their maximum willingness to pay and their net utility will be zero. But user i, $i \in H_{k^o}$ transmits data at a rate $x_i(\lambda^{(n)})$ and it is allocated a rate $x_i'(\lambda^{(n)}) = x_i(\lambda^{(n)})$, since $\sum_{i \in H_{k^o}} x_i(\lambda^{(n)}) < C$ by the conditions given in (12). Hence, it obtains positive net utility, since

$$\begin{aligned} 0 \;&=\; NU_i(\lambda_i^{max}, x_i(\lambda_i^{max})) \\ &=\; U_i(x_i(\lambda_i^{max})) - \lambda_i^{max} x_i(\lambda_i^{max}) \\ &<\; U_i(x_i(\lambda_i^{max})) - \lambda^{(n)} x_i(\lambda_i^{max}) \\ &\le\; NU_i^{max}(\lambda^{(n)}) \\ &=\; NU_i(\lambda^{(n)}, x_i(\lambda^{(n)})). \end{aligned}$$

Now, suppose that $\lambda^{(n)} < \lambda_{k^o}^{max}$. Then, user i, $i \in S_{k^o} \cup H_{k^o}$ transmits data at a rate $x_i(\lambda^{(n)})$ and it is allocated a rate $x_i'(\lambda^{(n)}) < x_i(\lambda^{(n)})$, since $\sum_{i \in S_{k^o} \cup H_{k^o}} x_i(\lambda^{(n)}) > C$ by conditions in (12). In this case, we can show that if $\lambda^{(n)}$ is close enough to $\lambda_{k^o}^{max}$ and users in set S_{k^o} transmit data, they obtain negative net utilities. Since $x_i(\lambda)$ is a continuous function of λ for $\lambda_{L_{k^o}}^{max} < \lambda \le \lambda_{k^o}^{max}$ and f_i is a continuous function of $\bar{x}$ by assumption (A3), $x_i'(\lambda)$ is a continuous function of λ for $\lambda_{L_{k^o}}^{max} < \lambda \le \lambda_{k^o}^{max}$. Thus, $NU_i(\lambda, x_i'(\lambda))$ is also a continuous function of λ for $\lambda_{L_{k^o}}^{max} < \lambda \le \lambda_{k^o}^{max}$. Moreover, for user i, $i \in S_{k^o}$,

$$\begin{aligned} 0 \;&=\; NU_i^{max}(\lambda_{k^o}^{max}) \\ &=\; U_i(x_i(\lambda_{k^o}^{max})) - \lambda_{k^o}^{max} x_i(\lambda_{k^o}^{max}) \\ &>\; U_i(x_i'(\lambda_{k^o}^{max})) - \lambda_{k^o}^{max} x_i'(\lambda_{k^o}^{max}) \\ &=\; NU_i(\lambda_{k^o}^{max}, x_i'(\lambda_{k^o}^{max})), \end{aligned} \tag{13}$$

where the inequality comes from the fact that $U_i(x_i(\lambda_{k^o}^{max}))$ is in the concave region by (R5). Since $NU_i(\lambda, x_i'(\lambda))$ is a continuous function for λ for $\lambda_{L_{k^o}}^{max} < \lambda \le \lambda_{k^o}^{max}$ and $NU_i(\lambda_{k^o}^{max}, x_i'(\lambda_{k^o}^{max})) < 0$, there exists a constant ϵ_i such that if $0 \le \lambda_{k^o}^{max} - \lambda \le \epsilon_i$, then user i, $i \in S_{k^o}$ obtains negative net utility (i.e., $NU_i(\lambda, x_i'(\lambda)) < 0$) when it transmits data at price λ. This implies that there exists $\epsilon_i > 0$ such that if $|\lambda - \lambda_{k^o}^{max}| \le \epsilon_i$, user i, $i \in S_{k^o}$, receives negative net utility by transmitting data. Further, since $\lambda^{(n)}$ converges to $\lambda_{k^o}^{max}$, there exists an m_i such that $|\lambda^{(n)} - \lambda_{k^o}^{max}| < \epsilon_i$ for all $n \ge m_i$. Hence, by the "self-regulating" property in assumption (A1), user i, $i \in S_{k^o}$ stops transmitting data after iteration m_i.

However, even if there exists iteration m_i after which user i, $i \in S_{k^o}$ receives negative net utility by transmitting data, it is not easy for the user to realize that moment. For example, during a transient period, the user may receive negative net utility, even though it may receive positive net utility in the future. Hence, it may not be a good strategy to stop transmitting data immediately after it receives negative net utility. Thus, the idea behind (A2) is to turn user i off not immediately, but only after it has received non-positive net utility for th_i consecutive iterations. This implies that, by appropriately choosing th_i, user i stops transmitting data only after th_i iterations of iteration m_i. After some users in the set S_{k^o} stop transmitting data, we can always have a situation such that

$$\sum_{i \in H_{k^o} \cup S_{k^o}^R} x_i(\lambda_{k^o}^{max}) < C, \tag{14}$$

where $S_{k^o}^R$ is a set of users that are still transmitting data among users in set S_{k^o}. Thus, for the remaining users in set $H_{k^o} \cup S_{k^o}^R$, we can find a $\lambda^* < \lambda_{k^o}^{max}$ such that

$$\sum_{i \in H_{k^o} \cup S_{k^o}^R} x_i(\lambda^*) \;=\; \sum_{i \in H_{k^o} \cup S_{k^o}^R} M_i \le C, \tag{15}$$

$$\text{or} \sum_{i \in H_{k^o} \cup S_{k^o}^R} x_i(\lambda^*) \;=\; C. \tag{16}$$

We can easily show that if (15) is satisfied, $\bar{x}(\lambda^{(n)})$ converges to $\bar{M}$, which is a global optimal rate allocation for the remaining users in set $H_{k^o} \cup S_{k^o}^R$. Also, if (16) is satisfied, by Proposition 3, the algorithm converges to the global optimal rate allocation for the remaining users in set $H_{k^o} \cup S_{k^o}^R$. Note that, in this scheme, it is important to have an appropriate threshold of tolerance, th_i. If it is too small, user i may stop transmitting data during the transient period, even if it can receive positive net utility in the future. On the other hand, if it is too large, the algorithm may take very long to converge.

As long as the users are "self-regulating", the proposed algorithm converges to the feasible rate allocation. Hence, our rate control algorithm does not cause congestion within the network even with non-concave utility functions. However, we still need to study its efficiency, since it may not result in a globally optimal rate allocation for all users, even though it results in an optimal rate allocation for the remaining users. Thus, in the following, we study the efficiency of the proposed rate allocation. We still assume that there exist subsets of users $H_{k^o,(N)}$, $S_{k^o,(N)}$, and $L_{k^o,(N)}$ that satisfy one of conditions in (12), since, otherwise, we know that our rate allocation is a global optimal rate allocation for all users. Further, assume that $S^R_{k^o,(N)}$ is the set of the remaining users in (14). Here, N is the number of users in the system.

First, We define the following variables.

- $\bar{x}^o_{(N)}$: the global optimal rate allocation.
- $\lambda^{max}_{k^o,(N)}$: the dual optimal solution.
- $\bar{x}_{(N)}(\lambda)$: the transmission rate at λ.
- $\bar{x}^*_{(N)}$: the proposed rate allocation.
- $\lambda^*_{(N)}$: an equilibrium price at the proposed rate allocation.

Then, the next proposition gives us an upper bound on the difference between the global optimal rate allocation and the proposed rate allocation.

Proposition 4: $\sum_{i=1}^{N} U_i(x^o_{i,(N)}) - \sum_{i=1}^{N} U_i(x^*_{i,(N)}) \leq \sum_{i \in S^C_{k^o,(N)}} U_i(x_{i,(N)}(\lambda^{max}_{k^o,(N)}))$, where $S^C_{k^o,(N)} = S_{k^o,(N)} - S^R_{k^o,(N)}$.

The next corollaries immediately follow from Proposition 4.

Corollary 1: If $\sum_{i=1}^{N} U_i(x^o_{i,(N)}) \rightarrow \infty$ and $\frac{\sum_{i \in S^C_{k^o,(N)}} U_i(x_{i,(N)}(\lambda^{max}_{k^o,(N)}))}{\sum_{i=1}^{N} U_i(x^o_{i,(N)})} \rightarrow 0$ as $N \rightarrow \infty$,
then $\frac{\sum_{i=1}^{N} U_i(x^*_{i,(N)})}{\sum_{i=1}^{N} U_i(x^o_{i,(N)})} \rightarrow 1$ as $N \rightarrow \infty$.

Corollary 2: If $\sum_{i=1}^{N} U_i(x^o_{i,(N)}) \rightarrow \infty$ as $N \rightarrow \infty$ and $\lambda^{max}_i \neq \lambda^{max}_j$ for $i \neq j$,
then $\frac{\sum_{i=1}^{N} U_i(x^*_{i,(N)})}{\sum_{i=1}^{N} U_i(x^o_{i,(N)})} \rightarrow 1$ as $N \rightarrow \infty$.

Corollaries 1 and 2 show the asymptotic optimality of our rate allocation. In other words, our rate allocation is a good approximation of a global optimal rate allocation when there are many users in a system with large capacity and the number of users in set $S^C_{k^o,(N)}$ is relatively small. Hence, for our algorithm to converge to an efficient rate allocation, we need the condition that the number of users in set $S^C_{k^o,(N)}$ is relatively small. We will study the effect that this condition has on the efficiency of our algorithm later and also propose some methods to make this number small.

Thus far, we have shown that the algorithm based on the subgradient and the "self-regulating" property converges to the feasible and asymptotically optimal rate allocation. As mentioned before, in the subgradient based algorithm, we cannot guarantee convergence with a constant step size. Hence, we use a step size that diminishes to zero. However, the constant step size can more efficiently track system variations, such as initiation and completion of calls than the diminishing step size. In the next proposition, we will show that if each user applies the "self-regulating" property, there exists a constant step size α for which the algorithm in (3) and (10) converges.

Proposition 5: With the "self-regulating" property of users, there exists a constant step size α with which the proposed algorithm converges.

E. Complexity

In this subsection, we compare the complexity of our subgradient based algorithm that considers both concave and sigmoidal-like utility functions with that of the gradient based algorithms in [4], [6] that consider only concave utility functions.

To calculate the price of the next iteration, the subgradient based algorithm uses a subgradient while the gradient algorithm uses a gradient. Further, in general, we cannot guarantee convergence of the subgradient algorithm with a constant step size, while the gradient based algorithm converges with a constant step size. However, in our algorithm, a subgradient is calculated from the difference between the capacity and the aggregate transmission rate of all users that use the node, which is similar to calculating a gradient in the gradient based algorithm. Moreover, in Proposition 5, we have shown that our algorithm converges even with a constant step size when each user is "self-regulating". Thus, our algorithm and the algorithms in [4], [6] have the same price update rule at the node. This implies that we need not modify the algorithm for concave utility functions at the node to allow sigmoidal-like utility functions.

Further, both of the algorithms have the same update rule for the transmission rate in each user, even though we require "self-regulating" property of users for convergence of our algorithm. This property is required because $x_i(\lambda)$ in (3) is not continuous at λ^{max}_i, if the utility function of user i is a sigmoidal-like function. However, if the utility function of user i is a concave function, $x_i(\lambda)$ is continuous and we do not need the "self-regulating" property for user i. Hence, compared with the algorithms in [4], [6], in our algorithm, we only have to add the "self-regulating" property to users with sigmoidal-like utility functions. This requires only calculating the received net utility by measuring the received rate. This can be easily done by counting the number of ACK packets or by explicit notification of the received rate from the destination.

F. The worst case

In the previous subsection, we have shown that the proposed rate allocation could be a good approximation of the global optimal rate allocation. However, it could also be inefficient in certain cases. Here, we show an example of the worst case and provide solutions to resolve it. We assume that each user i has the same utility function U that is a sigmoidal-like function and the same threshold of tolerance th. By assuming that each user has the same utility function, each user has the same maximum willingness to pay λ^{max}. Further assume that $\sum_{i=1}^{N} x_i(\lambda^{max}) > C$. In this case, $H_{k^o} = \emptyset$, $L_{k^o} = \emptyset$, and $S_{k^o} = \{1, 2, \cdots, N\}$. Moreover, all users in set S_{k^o} stop transmitting data at the same time (i.e., $S_{k^o}^C = \{1, 2, \cdots, N\}$), since all users have the same threshold of tolerance th. Hence, the system utility achieved by the proposed rate allocation will be zero.

To resolve this problem, we propose two solutions that attempt to make the number of users in set $S_{k^o}^C$ small. If the number of users in set $S_{k^o}^C$ is small, by Proposition 4, we can obtain an efficient rate allocation that is a good approximation of the global optimal rate allocation. First, we can slightly perturb (randomly) the utility function of each user. By doing this, each user i has a different maximum willingness to pay, λ_i^{max}, with high probability while making the effect on the performance of each user small. This makes the number of users in set S_{k^o} (and, thus, $S_{k^o}^C$) small with high probability, since users in set S_{k^o} have the same maximum willingness to pay $\lambda_{k^o}^{max}$. Second, we can assume that the threshold of tolerance of each user depends on the preference of the user. Thus, some users can tolerate negative net utility for a long time while some users can tolerate it for a short time. This makes users stop transmitting data at different iterations even if they have the same maximum willingness to pay. Hence, the number of users in set $S_{k^o}^C$ can be small.

IV. Numerical results

In this section, we provide simulation results using an ns-2 simulator. We consider a single bottleneck system in Fig. 2. In this figure, we provide the capacity and the propagation delay of each link. User i transmits packets from source node S_i to destination node D_i with utility function U_i. Users 1 and 4 have a sigmoidal utility function defined as

$$U_i(x) = c_i\left(\frac{1}{1+e^{-a_i(x-b_i)}} + d_i\right),$$

where c_i and d_i are used for the normalization of the function and x is a rate in a unit of Megabit per second (Mbps). Users 2 and 3 have a log utility function defined as

$$U_i(x) = c_i(\log(a_i x + b_i) + d_i).$$

In this simulation, we normalize the utility function such that $U_i(0) = 0$ and $U_i(M_i) = 1$, where M_i is the maximum transmission rate of user i (note that it is not necessary to normalize the utility function). User i has its threshold of tolerance, th_i and starts transmitting data packets at time st_i sec. We provide parameters for each user in Table I and plot the utility function of each user in Fig. 3.

The node updates its price per unit rate every 200 msec using (10) with a constant step size of 0.03. To forward the price to users, we add a field for the price in the header of a packet. Whenever a packet passes through a node, the node writes its current price in the field and it is sent to the destination. At the destination, the price in the received packet is copied to the field of an ACK packet and is sent to the source. We assume that a data packet and an ACK packet consist of 500 bytes and 40 bytes, respectively. The source estimates the received rate by counting the number of ACK packets and calculates the received utility and the received net utility by using the estimated received rate. By the transmission rate update rule, if the price becomes higher than its maximum willingness to pay, a user does not transmit data packets. However, if the user does not transmit data packets during the transient period, it cannot be informed of the price for the next iteration, since the price is conveyed by ACK packets from the destination in our simulation setting. Thus, we allow the user to transmit packets at a very low rate, even though its transmission rate that maximizes its net utility is zero during the transient period. By doing this, the user can be informed the price for the next iteration by the ACK packets from the destination. To that end, in the simulation, a user transmits two packets that consist of 40 bytes, every iteration (200 msec).

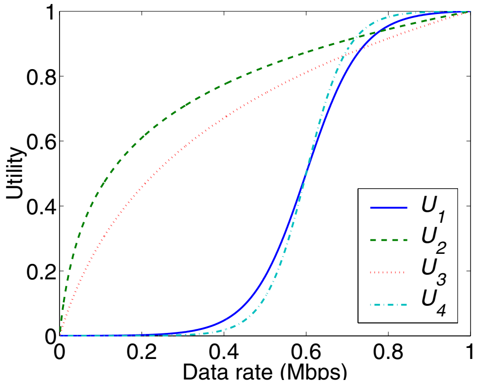


Fig. 3. The utility function of each user.

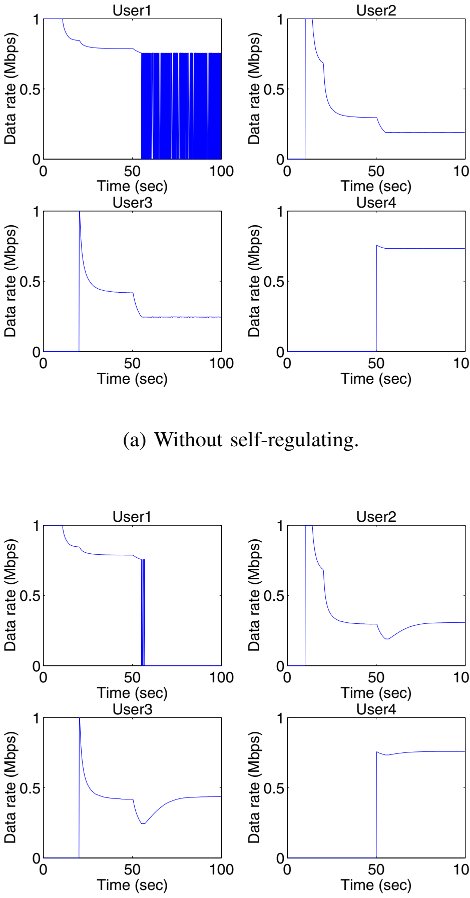
We compare two systems: a system with the "self-regulating" property and a system without the "self-regulating" property. Note that the algorithm for the system without the "self-regulating" property is the same as the gradient based algorithms in [4], [6]. Thus, the results for this system show the behavior of the algorithms developed in the literature for concave utility functions when applied to a network supporting users with both concave and sigmoidal utility functions. We plot the transmission rate, the received rate, and the received net utility of each user in Figs. 4, 5, and 6, respectively. We also compare the variation of the price per unit rate of each system in Fig. 7.

The results show that before user 4 starts transmitting packets (50 sec), the two systems yield the same results. When only users 1, 2, and 3 are in the system, as shown in Table I, $x_1(\lambda_1^{max}) + x_2(\lambda_1^{max}) + x_3(\lambda_1^{max}) = 1.191$ (Mbps) $<$ 1.5 (Mbps), where λ_1^{max} is the smallest maximum willingness to pay among those of users in the system. Thus, we can have $\lambda^o < \lambda_1^{max}$ such that $x_1(\lambda^o)+x_2(\lambda^o)+x_3(\lambda^o) = 1.5$ (Mbps)

TABLE I

PARAMETERS FOR USERS.

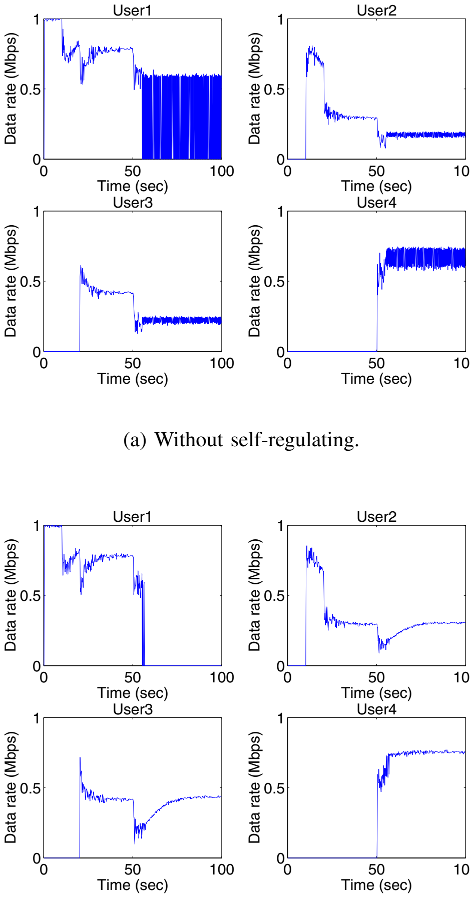
User i	Type	a_i	b_i	M_i	th_i	st_i	λ_i^{max}	$x_i(\lambda_1^{max})$	$x_i(\lambda_4^{max})$
1	Sigmoid	15	0.6	1	20	0	1.210	0.756	0
2	Log	50	1	1	20	10	12.717	0.190	0.179
3	Log	10	1	1	20	20	4.170	0.245	0.226
4	Sigmoid	20	0.6	1	20	50	1.276	0.734	0.731



(a) Without self-regulating.

(b) With self-regulating.

Fig. 4. Transmission data rate.



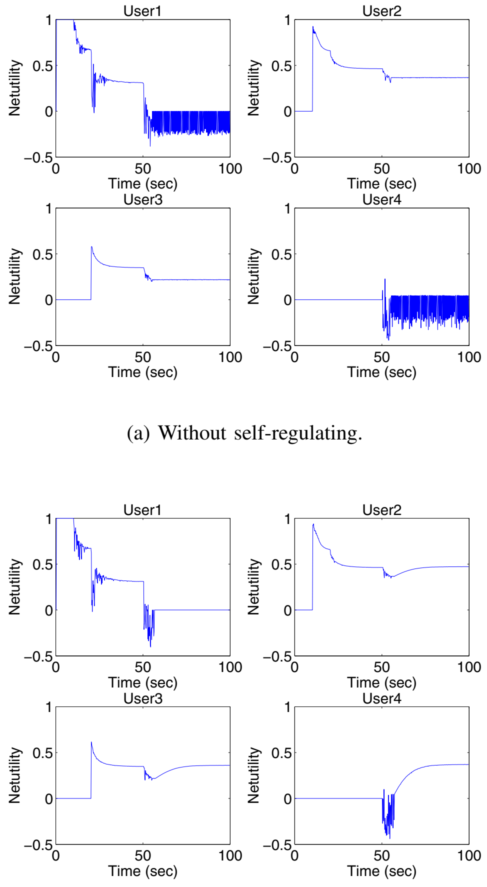
(a) Without self-regulating.

(b) With self-regulating.

Fig. 5. Received data rate.

that satisfies the condition in Proposition 3 and the algorithm converges to the optimal rate allocation without relying on the "self-regulating" property of users.

However, when all four users are in the system, as shown in Table I, $x_1(\lambda_1^{max}) + x_2(\lambda_1^{max}) + x_3(\lambda_1^{max}) + x_4(\lambda_1^{max}) =$ 1.925 (Mbps) $>$ 1.5 (Mbps) and $x_2(\lambda_1^{max}) + x_3(\lambda_1^{max}) +$ $x_4(\lambda_1^{max}) =$ 1.169 (Mbps) $<$ 1.5 (Mbps), where λ_1^{max} is the smallest maximum willingness to pay among users. Thus, there is no λ^o such that $x_1(\lambda^o) + x_2(\lambda^o) + x_3(\lambda^o) + x_4(\lambda^o) =$ 1.5 (Mbps) and the condition in Proposition 3 is not satisfied. Therefore, in the system without the "self-regulating" property, after user 4 starts transmitting packets, the transmission rate of user 1 (the primal solution) keeps oscillating, as shown in Fig. 4(a), even though the price (the dual solution) in Fig. 7 converges to around $\lambda_1^{max} = 1.210$ (the dual optimal solution), as proven in Proposition 3. In this case, when user 1 transmits packets, the aggregate transmission rate of all users exceeds the capacity of the node. This causes congestion at the node and a large number of losses of packets for all users. Thus, as shown in Figs. 4(a) and 5(a), each user has a large difference between the transmission rate and the received rate. Further, due to these packet losses, some users have negative received net utility, even though each user determines its transmission rate by (3) so that it has non-negative net utility assuming that there is no packet loss. As shown in Fig. 6(a), after user 4 starts transmitting packets, the net utility of user 1 becomes



(a) Without self-regulating.

(b) With self-regulating.

Fig. 6. Received net utility.

non-positive and the net utility of user 4 oscillates between positive and negative values. These results show that if there exist users with non-concave utility functions in the system, using a rate control algorithm devised only for concave utility functions could result in an unstable system as well as a large amount of network congestion.

However, in the system with the “self-regulating” property, as shown in Fig. 4(b), user 1 stops transmitting packets due to the “self-regulating” property, after having received non-positive net utility values for th_1 consecutive iterations. After user 1 stops transmitting packets, as shown in Table I, $x_2(\lambda_4^{max}) + x_3(\lambda_4^{max}) + x_4(\lambda_4^{max}) = 1.136$ (Mbps) $<$ 1.5 (Mbps), where λ_4^{max} is the smallest maximum willingness to pay among those of users that remain in the system. Thus, we can have $\lambda^* < \lambda_4^{max} = 1.276$ such that $x_2(\lambda^*) + x_3(\lambda^*) + x_4(\lambda^*) = 1.5$ (Mbps). This satisfies the condition in Proposition 3 and the algorithm converges to the optimal rate allocation for the remaining users. In this case, the aggregate transmission rate for users converges to the capacity of the node (1.5 Mbps). Thus, as shown in Figs. 4(b) and 5(b), the transmission rate of each user converges and the received rate of each user is almost same as its transmission rate. This implies that with the “self-regulating” property, the system stabilizes and congestion is alleviated.

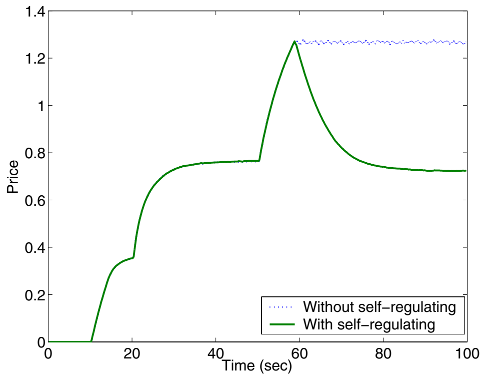


Fig. 7. Price.

In these results, we must note that user 1 that has already been in the system stops transmitting packets by the arrival of a new user (user 4). It may be undesirable to interrupt existing services.[2] However, recall that, in this paper, we consider the situation that is similar to the current Internet in which the system does not have a central authority for call admission control and rate control, and a user adjusts its transmission rate according to a congestion indicator from the system without considering the other users. In such a situation, as shown in the results, by continuing to transmit packets, user 1 has negative net utility value as well as a large number of packet losses that might be unsatisfactory to the service. Therefore, it may be beneficial not only to the other users but also to user 1 itself for it to stop transmitting packets. User 1 may restart its transmission after some random time[3].

The results also tell us the following. First, a service with a concave utility function can be better adapted to congestion on the link than a service with a sigmoidal-like utility function. The former can adjust its transmission data rate gradually according to the congestion level on the link, while the latter can adjust its transmission rate gradually only up to a certain level. Further, the former has a higher degree of adaptation to the level of the congestion than the latter. This implies that by modeling traditional data services with concave utility functions and real-time streaming services as sigmoidal-like utility functions, we can exploit the characteristics of each service appropriately. Second, from the pricing point of view, if a real-time service with a sigmoidal-like utility function wants to have a higher priority to be served than a data service with a concave utility function, it must have a higher maximum willingness to pay than the data service. In this case, in general, the real-time service pays more for the service than the data service, since the real-time service

[2] This happens because of the property of utility and pricing based algorithms. Hence, this may happen even in the system in which all users have concave utility functions, if users do not have the minimum rate that must be guaranteed or their maximum willingness to pays are not infinity.

[3] Finding a good strategy for this will be a topic for future research.

keeps transmitting packets even though the data services stop transmitting because of the high price. This implies that the real-time service must be more expensive than the data service. Thirdly, when a new service enters into the network, it may be inevitable to interrupt existing services to preserve the system efficiency without incurring congestion. Hence, to prevent this from happening, the system should have an appropriate, preferably distributed, call admission control that admits a new service if it does not interrupt existing (real-time) services.

V. Conclusion

In this paper, we have studied the distributed rate control algorithm by considering both sigmoidal-like and concave utility functions. We have shown that in the presence of sigmoidal-like utility functions, an algorithm that converges to an efficient rate allocation for a system with only concave utility functions, may not converge, exhibiting oscillatory behavior. Further, such algorithms may result in excessive congestion within the network. This implies that rate control algorithms that have been developed only for concave functions might be inefficient in more realistic settings. To overcome these difficulties, we have developed a distributed algorithm where each user has a "self-regulating" property. Our algorithm works for both sigmoidal-like and concave utility functions. We have shown that our algorithm converges to the asymptotically optimal rate allocation and that its complexity is comparable to that of algorithms developed only for concave utility functions.

In this paper, we have focused on the study of the stability and the optimality issues of the non-convexity in rate control considering a system with a single bottleneck system. In future work, we will investigate the implementation of the algorithms in a network with multiple bottleneck links. Here, the issues are on how to aggregate the prices from multiple bottleneck links to provide users with the signal to self-regulate.

References

[1] S. Shenker, "Fundamental design issues for the future Internet," *IEEE journal on selected area in communications*, vol. 13, pp. 1176–1188, 1995.

[2] F. P. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, pp. 33–37, 1997.

[3] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, pp. 237–252, 1998.

[4] S. H. Low and D. E. Lapsley, "Optimization flow control-I: basic algorithm and convergence," *IEEE/ACM Transactions on Networking*, vol. 7, pp. 861–874, 1999.

[5] S. Athuraliya and S. H. Low, "Optimization flow control-II: Implementation," submitted for publication.

[6] H. Yäiche, R. R. Mazumdar, and C. Rosenberg, "A game theoretic framework for bandwidth allocation and pricing of elastic connections in broadband networks: theory and algorithms," *IEEE/ACM Transactions on Networking*, vol. 8, pp. 667–678, 2000.

[7] S. Kunniyur and R. Srikant, "End-to-end congestion control schemes: utility function, random losses and ECN marks," in *IEEE Infocom'00*, 2000, pp. 1323–1332.

[8] R. J. La and V. Anantharam, "Utility-based rate control in the Internet for elastic traffic," *IEEE/ACM Transactions on Networking*, vol. 10, pp. 272–286, 2002.

[9] K. Kar, S. Sarkar, and L. Tassiulas, "A simple rate control algorithm for maximizing total user utility," in *IEEE Infocom'01*, 2001, pp. 133–141.

[10] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Transactions on Networking*, vol. 8, pp. 556–567, 2000.

[11] J.-W. Lee, R. R. Mazumdar, and N. B. Shroff. (2003) Non-convexity issues for Internet rate control with multi-class services: stability and optimality. Technical Report, Purdue University. [Online]. Available: http://expert.cc.purdue.edu/∼lee46/Documents/src.pdf

[12] J. W. Lee, R. R. Mazumdar, and N. B. Shroff, "Downlink power allocation for multi-class CDMA wireless networks," in *IEEE Infocom'02*, 2002, pp. 1480–1489.

[13] N. Z. Shor, *Minimization methods for non-differentiable functions*. Springer-Verlag, 1985.

[14] M. Minoux, *Mathematical programming:theory and algorithms*. Wiley, 1986.

[15] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1999.

0-7803-8356-7/04/$20.00 (C) 2004 IEEE