Proceedings of the 38th
Conference on Decision & Control
Phoenix, Arizona USA • December 1999

Network Pricing Using Game Theoretic Approach

Richard J. La, and Venkat Anantharam
Electrical Engineering & Computer Sciences, U.C. Berkeley
hyongla@eecs.berkeley.edu, ananth@eecs.berkeley.edu

Abstract

In this paper we model the interaction between the network and the network users as a noncooperative game and study the Nash equilibrium points (NEP) of the game. We show that there is a unique NEP of the game in our model with generalized power functions as the benefit function, and that the Nash mapping from the price vector space to the feasible system flow configuration space is both continuous and injective. We discuss how the network, an active player, can drive the users into an efficient state, by using a pricing mechanism. The convergence result for both synchronous and asynchronous schemes for a simple network is proved when users adopt a variation of the greedy algorithm.

We also demonstrate that the results proved in this paper depend on the natural properties of the benefit functions, not on the particular form of the benefit functions used in the paper. This tells us that the results proven in this paper can be applied to more general benefit functions.

1 Introduction

In the future network services are likely to be provided by profit-seeking, private entities. These network service providers (NSP) may provide different QoS services at various prices. In order to recover their costs and raise profits, these NSPs need to understand the users' behavior. If network users must pay for their service, each network user will be interested only in maximizing its own benefit at the lowest possible cost, and its choice of action will depend on those of other users. The behavior of users in such an environment can be addressed in the framework of game theory. One fundamental question that arises in such an environment is the existence and possibly uniqueness of an equilibrium where no user finds it beneficial to change its flow configuration unilaterally. Such an equilibrium is called a Nash equilibrium point (NEP) [1].

We consider a simple network with a finite number of users. The network service provider presents to the users a menu that describes the capacities of the servers and the price per unit flow for each server. Users decide their flow rates to each server after they see the menu and agree to pay the network. Each user receives some benefit from obtaining a certain amount of flow rate, which depends on the delay it experiences. Some users may be more sensitive to delays than others, and some may get higher benefit than others for receiving the same amount of flow rate. If users are charged for the amount of flow rate they send, their choices of flow rates will depend not only on the actions of other users but also on the prices they have to pay to the network. The users are assumed to be selfish and interested only in maximizing their own benefit. We model this interaction between network and users and among the users as a noncooperative game and show the existence of an NEP and its uniqueness. We also show that the Nash mapping from the price vector space to the system flow configuration space is continuous and injective.

The fact that the flow rates chosen by the users depend on the prices suggests that the NSP may be able to control the network load through pricing. In the second part of the paper we investigate the case where the network is an active player that tries to maximize its objective function. The network initially observes the actions chosen by the users and attempts to estimate the parameters of the users. Based on the estimated net benefit functions of the users, the network exercises price discrimination to drive the users to an operating point that maximizes its objective function. Convergence results for a simple network are demonstrated under the Gauss-Seidel and Jacobi schemes. We show that the uniqueness and convergence results depend on the natural properties benefit functions would possess, but not so much on the form of the benefit functions used in this paper. This tells us that the results proved in this paper can be extended to much more general benefit functions.

The rest of the paper is organized as follows. Section 2 describes the model, and section 3 shows the existence and uniqueness of an NEP, which is followed by the proof of the continuity and injectivity of the Nash mapping from the price vector space to the feasible system flow configuration space. Then, we discuss the case of a more active network that attempts to drive the users to its optimal point, and prove the convergence results. This is followed by a discussion on the robustness of NEPs. A numerical example is given in section 7.

2 Model and Problem Formulation

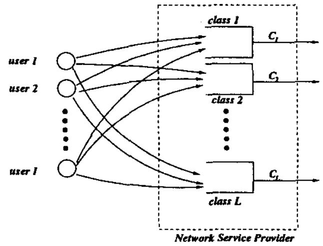

Figure 1: Network Model.

We consider a simple network model as shown in Figure 1. There are a finite number of users, $I = \{1, 2, ..., \mathrm{I}\}$, that share the network. The network service provider (NSP) provides different classes of service with various service rates. Let $L = \{1, 2, ..., \mathrm{L}\}$ be the set of classes of service and $C_l, l \in L$, be the service rate of server l if there is a dedicated server for each class, or the service rate allocated to class l if there is only one server. In the second case all the residual service rate can be assumed to be diverted to the best-effort service class with no guaranteed service rate. We assume that each class has a dedicated server. Without loss of generality, we assume that classes are ordered by decreasing service rate, i.e., $C_1 \geq C_2 \geq \cdots \geq C_{\mathrm{L}}$. Each user is assumed to be selfish in the sense that it only attempts to maximize its own net benefit.

The NSP presents to the users a menu with prices per unit flow of each server and the service rates of the servers. The price per unit flow of server $l \in L$ that user i needs to pay, denoted by $p_l^i \geq p_{min} > 0$, is a positive real number. The minimum price per unit flow p_{min} charged by the network can be made arbitrarily small. Our formulation includes the special case where the network does not discriminate among the users, and charges the same price $p_l = p_l^i$ for all $i \in I$. In this paper we study more general cases where price discrimination is allowed. Hence, the price vector p is an element in $\mathcal{P} = \mathcal{R}^{\mathrm{I} \times \mathrm{L}}$, where $\mathcal{R} = [p_{min}, \infty)$. User i sees the menu and decides how much flow rate r_l^i it wants from each server l and agrees to pay $\sum_{l \in L} p_l^i \cdot r_l^i$ to the network. We must have $r_l^i \geq 0$ (nonnegativity constraint) and $r_l^i \leq C_l$ (service rate constraint) because the users know the service rate of each server l before they make the decision. Let r_l denote the total flow rate to server l from the users. The flow rate configuration of user i is denoted by r^i, and the system flow configuration by $r = (r^1, r^2, ..., r^{\mathrm{I}})$. Let $r^{-i} = (r^1, ..., r^{i-1}, r^{i+1}, ..., r^{\mathrm{I}})$, $t^i = \sum_{l \in L} r_l^i$ denote user i's total flow rate, and t_l^{-i} the total class l flow rate from the users other than user i. A user flow rate configuration r^i is said to be feasible if it satisfies the nonnegativity and service rate constraints. We denote the set of all feasible flow rate configurations for user i by R^i. Note that R^i is same for every user. Similarly, a network flow rate configuration r is feasible if every user flow configuration is feasible, and $R = R^1 \bigotimes \cdots \bigotimes R^{\mathrm{I}}$ denotes the set of all feasible system flow configurations.

The net benefit user i receives when the system flow configuration is r is quantified by a function $J^i(r, p)$. This equals the benefit received from the flow minus the price it has to pay to the network to carry the flow. The benefit of user i depends not only on its own flow configuration but also on those of other users, and therefore, so does the net benefit. Since we assume that every user is selfish, the problem can be modeled as a noncooperative game where each user tries to maximize its net benefit. A natural question that arises in such settings is the existence of a Nash equilibrium point. In other words, we are interested in finding a system flow configuration such that no user finds it beneficial to change its own flow rate configuration when no other users do. Mathematically, a system flow configuration $\tilde{r} = (\tilde{r}^1, \tilde{r}^2, ..., \tilde{r}^{\mathrm{I}})$ is an NEP if, for all $i \in I$, the following holds:

$$\begin{aligned} J^i(\tilde{r}, p) &= J^i(\tilde{r}^1, ..., \tilde{r}^{i-1}, \tilde{r}^i, \tilde{r}^{i+1}, ..., \tilde{r}^{\mathrm{I}}, p) \\ &= \max_{r^i \in R^i} J^i(\tilde{r}^1, ..., \tilde{r}^{i-1}, r^i, \tilde{r}^{i+1}, ..., \tilde{r}^{\mathrm{I}}, p) \quad (1) \end{aligned}$$

The importance of an NEP is that it is a point at which no user has an incentive to deviate.

The network attempts to fit the net benefit functions of the users to a form given by

$$\begin{aligned} J^i(r, p) &= \sum_{l \in L} J_l^i(r, p) \\ &= \sum_{l \in L} ((r_l^i)^{\alpha_i} (C_l - \min\{r_l, C_l\})^{\beta_i} - r_l^i p_l^i \\ &\quad - K_i t^i r_l^i) \\ &= \sum_{l \in L} ((r_l^i)^{\alpha_i} (C_l - \min\{r_l, C_l\})^{\beta_i} - r_l^i p_l^i) \\ &\quad - K_i (t^i)^2 \qquad (2) \end{aligned}$$

where K_i is a fixed constant that governs the interdependency among user i's flow rates to the servers, and $\alpha_i \in (\alpha_0, 1), 0 < \alpha_0 < 1$, and $\beta_i \in (\beta_0, 1], 0 < \beta_0 < 1$, are the parameters that determine user i's sensitivity to the flow rate it receives and the delay it experiences. Here we assume that $K_i > 0$ for all $i \in I$ because we are mainly interested in the case where the flow rate of a user to a server also affects the benefit the user receives from another server. The first term in (2) is a generalized power function, and the second term is the price user i pays to the network. The last term accounts for the decrease in the rate at which the benefit increases as t^i increases, i.e., a decrease in marginal benefit with t^i [1]. The results we prove hold when the actual net benefit functions of the users are close to this form and satisfy certain natural qualitative properties.

[1] This is the law of diminishing utility.

3 Nash Equilibrium Point and Mapping

3.1 Nash Equilibrium Point

The following theorem proves the existence and uniqueness of NEP. The proof of this theorem is in [2].

Theorem 1 *There exists a Nash equilibrium of the game, and the Nash equilibrium is unique.*

3.2 Nash mapping

In the previous subsection we have proved the uniqueness of NEP. Consider the function $\mathcal{N} : \mathcal{P} \to R$, that assigns to each price vector $p \in \mathcal{P}$ the Nash equilibrium $\mathcal{N}(p)$ of the respective game. The function $\mathcal{N}$ will be referred to as the Nash mapping. In this section we show that the Nash mapping is both continuous and injective. The proof of this lemma is in [2].

Lemma 1 *The Nash mapping $\mathcal{N} : \mathcal{P} \to R$ is both continuous and one-to-one, i.e., if $\mathcal{N}(\tilde{p}) = \mathcal{N}(\bar{p})$, then $\tilde{p} = \bar{p}$.*

The Nash mapping is, however, not surjective. To see this one can show that there does not exist a price vector that yields an NEP $r = (C_1, ..., C_L)$. Hence, the image $\mathcal{N}(\mathcal{P})$ is a strict subset of R.

The fact that the Nash mapping $\mathcal{N}$ is injective has an important implication. Suppose that the NSP chooses a desired load level for each server based on requirements such as average delay guarantees, and the desired load is in $\mathcal{N}(\mathcal{P})$. Then, using the inverse mapping, the NSP can compute the unique price vector $p \in \mathcal{P}$ such that $\mathcal{N}(p)$ coincides with the desired load vector. For example, suppose that the network provides differentiated services with quality of service (QoS) guarantees and the average delay can be computed as a function of the total load at the server. Then, the network should control the load level at each server by computing the appropriate prices, at whose NEP the guaranteed QoS requirements are met. If users converge to an equilibrium, they will converge to an NEP, which is the desired load. This is discussed in sections 4 and 5.

4 Pricing as Congestion Control Mechanism

In the previous sections we have assumed that the network chooses the prices at the beginning and does not change them. In other words, after it sets the prices, the network makes no attempt to drive the users to a more preferable state for itself. In many cases, however, this may not be the case. For instance, the network, as a social player, might want to maximize the total benefits of the users. In some other cases, if the network is operated by a private entity, the objective of the network might be maximizing the total revenue. This suggests by setting prices we allow the network to be an active player in the game and investigate the possibility of achieving a more efficient state of the network. In this section we heuristically describe how this may be done. Formal results for simple cases are given in section 5.

Consider the following. The network does not a priori know the benefit functions of the users. Thus, the network cannot simply run an optimization algorithm to find the price vector that yields its maximum benefit at the unique NEP corresponding to the price vector. Hence, initially the network sets the price vector according to some rules and observes how users update their flow rates as described below.

Suppose that when a user gets a chance to update its strategy, it computes the unique [2] strategy that maximizes its net benefit function and updates its flow rates. We assume that when a user sees that a server is saturated after step n, i.e., $r_l(n) \geq C_l$, the user sets its flow rate to the server to zero at the next step. This is to ensure that the total rate to a server is strictly less than its capacity after a finite number of updates from any initial flow configuration. We call this the *greedy algorithm*. Under the greedy algorithm, the network can estimate the parameters α_i, β_i, and K_i of user i's net benefit function as follows. The network keeps records of the strategies used by the users when they update their flow rates. If each user updates at least three times and the available service rates to the user are different at each time, then the network can compute user i's parameters, α_i, β_i, and K_i that satisfy the three independent sets of nonlinear equations that result from the Kuhn-Tucker conditions [4] and determine its net benefit function or at least estimate it. If users' true net benefit functions are only approximate of the form in (2), then the network can continue to update the information regarding the true net benefit functions of the users until it has the approximates of the form in (2) that are closest to the true net benefit functions. We may assume that this estimation scheme converges, perhaps using some stopping rule.

Once the network knows the true net benefit functions of the users or at least has good estimates of the parameters, it can compute the price vector that optimizes the network objective or satisfies the QoS guarantees at the corresponding NEP. A numerical example is given in section 7.

After computing the price vector, the network informs the users of a price change and presents the new price

[2] The uniqueness of maximizing strategy follows from that the user faces a concave optimization problem given the strategies of other users.

vectors. Again, each user looks at the price presented by the network and decides its own flow rates to the servers. As described before, each user continues to update its flow rates to maximize its net benefit, given the aggregate flow rates of other users. One can imagine that the network and the users repeat this process.

Since the users do not necessarily start from the unique NEP, the above model is useful only if the users eventually reach the NEP, i.e., they converge to the NEP in the limit. If the users do not ever reach a point close to the NEP, then the network does not have any reason for computing the price vector that maximizes its benefit at the corresponding NEP. This is discussed in the next section.

One expects that if the users converge to an NEP (with the actual net benefit functions but with prices set by the network based on its estimated net benefit functions), it should be close to the NEP expected by the network. This is discussed in section 6.

5 Stability of NEP

In this section, we address the stability of the NEPs in more details. We only discuss the simple cases with only one server. Suppose that the network sets the prices and the users iteratively update their flow configurations in a self-optimizing way. We show that there is convergence to the NEP under both synchronous and asynchronous Gauss-Seidel schemes. Due to an oscillation effect, convergence is not guaranteed under a Jacobi scheme. However, a modified Jacobi scheme with a damping constant is proved to yield convergence. Although it is not discussed here, the Gauss-Seidel scheme can be proved to converge in the two-server and two-user cases also.

5.1 The Gauss-Seidel Scheme

In this subsection we consider a dynamic scheme in which users update their flow rates in an asynchronous manner. In other words, the users change their flow rates in a sequence in such a way that in each step only one user updates its flow rate. We assume that each user has an accurate measurement of the aggregate flow rate before updating its flow rate. Although we do not assume any particular order of updates, we require that there is some finite number that upper bounds the number of steps between any two subsequent updates of each user. This is called an asynchronous Gauss-Seidel scheme.

We assume that the price vector $p = (p_1, ..., p_I)$, where p_i is the price charged to user i, is fixed by the network and let $r^* = (r^{1*}, ..., r^{I*})$ and t^* denote the unique NEP corresponding to p and the total flow rate at the NEP r^*, i.e., $t^* = \sum_{i=1}^{I} r^{i*}$, respectively. Assume that $r(0)$ is such that $t(0) < C$, where C is the available service rate of the server. This guarantees that $t(n) < C$ for all n because the optimal flow rate for the updating user is always strictly small than the service rate available to the user. The following theorem states that the users' flow rates converge to the unique NEP under the Gauss-Seidel scheme. This theorem is proved in [2].

Theorem 2 *Under the asynchronous Gauss-Seidel scheme, the flow rate vector $r(n)$ converges to the unique NEP r^* as $n \to \infty$, i.e., $lim_{n\to\infty} r(n) = r^*$.*

Since the synchronous Gauss-Seidel scheme is a special case of the asynchronous Gauss-Seidel scheme, the synchronous Gauss-Seidel scheme also converges.

5.2 The Jacobi Scheme

This subsection discusses the convergence of a synchronous dynamic scheme, called the modified Jacobi scheme. Under the Jacobi scheme, at each step n all users update their flow rates simultaneously, based on the information on the total flow rate from the previous step, $n-1$. As stated in [3] this assumption on simultaneous update should be interpreted as that each user has available only information from the previous step due to delays, rather than as a synchronization requirement. We show that if users use a modified version of the greedy algorithm, then the flow rate vector $r(n)$ converges to the unique NEP. Suppose that $t(n) < C$, where C is the service rate of the server, and the users adopt the following updating rule

$$r^i(n+1) = r^i(n) + \frac{1}{I}(\hat{r}^i(n+1) - r^i(n)), \tag{3}$$

where I is the number of users[3] and $\hat{r}^i(n+1)$ is the flow rate computed by user i according to the original greedy algorithm. We assume that $t(0) < C$. This ensures that $t(n) < C$ for all $n \geq 0$. The following theorem states that the users' flow rates converge to the NEP under the modified Jacobi scheme. The proof is in [2].

Theorem 3 *Under the modified Jacobi scheme given in (3),*

$$lim_{n\to\infty}\ r(n) = r^*.$$

6 Robustness of NEP

In the previous sections we have assumed that the net benefit function of each user is of the form given in (2). The true net benefit function of a user, however, may or may not be of the form. In this section we show that the existence and uniqueness of NEP do not depend on the

[3]In fact, any constant M $\geq$ I yields convergence.

particular form of the net benefit function used in the paper, but are consequences of certain natural properties benefit functions are expected to possess. Suppose that users' benefit functions satisfy the following conditions.

1) Benefit functions are continuous.
2) The total benefit $V^i(r)$ of user i is given by the sum of benefits it receives from each server, i.e.,

$$V^i(r) = \sum_{l \in L} V_l^i(r) = \sum_{l \in L} V_l^i(r_l, r_l^i, t_{-l}^i)$$

where $t_{-l}^i = \sum_{l' \neq l} r_{l'}^i$. For fixed r_l^{-i} and t_{-l}^i, $V_l^i(r_l, r_l^i, t_{-l}^i)$ is maximized by some $r_l^i < C_l - r_l^{-i}$ if $r_l^{-i} < C_l$. Further, if $r_l \geq C_l$, user i receives at most zero benefit from a positive flow rate to server l. The net benefit $U^i(r,p)$ is given by

$$U^i(r,p) = \sum_{l \in L} U_l^i(r,p) = \sum_{l \in L} (V_l^i(r) - p_l^i \cdot r_l^i).$$

3) For all $i \in I$, given any r^{-i} such that $r_l^{-i} < C_l$ for some $l \in L$, benefit function is twice differentiable on $R^{i*} = \{r^i \in R^i \mid r_l^i < C_l - r_l^{-i}$ for all $l \in L$ such that $r_l^{-i} < C_l$ and $r_l^i = 0$ for all $l' \in L$ such that $r_{l'}^{-i} \geq C_{l'}$ $\}$. Further, it is strictly concave on $\bar{R}^{i*} = \{r^i \in R^i \mid$ $r_l^i \leq C_l - r_l^{-i}$ for all $l \in L$ such that $r_l^{-i} < C_l$ and $r_l^i =$ 0 for all $l' \in L$ such that $r_{l'}^{-i} \geq C_{l'}\}$.
4) The derivatives of the net benefit function defined on R^{i*}

$$K_l^i(r,p) = \frac{\partial U^i(r,p)}{\partial r_l^i} = K_l^i(r_l, r_l^i, t_{-l}^i, p_l^i)$$

satisfy the following properties:
(i) $K_l^i(r,p)$ is strictly decreasing in each of r_l^i, r_l, and t_{-l}^i if $r_l < C_l$. For instance, if r_l^i and r_l^{-i} are fixed, then $K_l^i(r,p)$ is decreasing in t_{-l}^i.
(ii) $K_l^i(r,p)\big|_{\tilde{r}} > K_l^i(r,p)\big|_{\bar{r}}$, where $\bar{r} = \tilde{r} + \delta \cdot (e_l^i - e_l^j), j \neq i$, and e_l^i is a unit vector whose only non-zero element is (i,l)-th element.

Let us first briefly motivate these assumptions. In order to adjust its flow rates based on its total flow rate and the delays, each user should be able to quantify the net benefit it receives from each server. This is captured by assumption (2). Furthermore, as a server becomes saturated, the delay will be intolerably large and users will receive no benefit from the server and may even be better off not sending the packets to the server at all. Therefore, at an equilibrium, if there exists any, the total flow rate to a server should be strictly smaller than its capacity. Although user i's benefit from server l increases with its flow rate r_l^i if the delay is fixed, the actual increase decreases with the total delay its flow experiences, which increases with the total flow rate. This naturally gives a rise to a concave benefit function. Assumption (4) says that the marginal benefit user i receives for extra ε rate to server l, decreases not only with r_l^i, but also with the total flow rate it sends to the other servers. This can be explained from the law of diminishing utility. Moreover, the marginal benefit decreases with the expected delay of the server, which depends on the total rate to the server r_l. Assumption $(4ii)$ is another way of stating that user i's marginal benefit decreases with r_l^i and t^i when r_l is fixed.

From the above assumptions, the existence of an NEP follows from [5]. It turns out that the uniqueness of NEP in the previous model follows from the properties these net benefit functions possess, and there is a unique NEP of the game with any net benefit functions satisfying 1)-4). Further, the Nash mapping defined in section 3 is continuous with such net benefit functions. The Nash mapping in this case, however, may or may not be injective. However, one can show that there are no two price vectors that yield the same NEP $\tilde{r}$ such that $\tilde{r}_l^i > 0$ for all $i \in I$ and $l \in L$.

Suppose that we use the following metric to measure the distance between net benefit functions :

$$d(U^i(\cdot), J^i(\cdot)) = \sup_{r \in R} \sum_{l \in L} \left| U_l^i(r,p) - J_l^i(r,p) \right|. \qquad (4)$$

Suppose now that the net benefit function of each individual user satisfies 1)-4) and is close to a net benefit function of the power function type described in section 2 according to (4). Suppose the network attempts to fit parameters (α_i, β_i, K_i) to describe the net benefit function of user i. We assume that the network uses a scheme that converges so that the parameters have been fit so as to give, for each user, a power function type net benefit function that is close to its actual net benefit function. The network now uses price discrimination to set the prices p_l^i based on its estimated net benefit functions.

We can show that the flow configuration of the users converges to the unique NEP of the game (with the actual net benefit functions of the users and the prices set by the network) under the Gauss-Seidel and also under the modified Jacobi scheme in a single server case[4]. Simulation results also indicate that when the interdependency between the benefits from different servers is not too large, users converge to the NEP even in multiple server cases. A numerical example is given in the next section. Furthermore, this NEP results in an overall benefit to the network that is close to the optimum that the network could assure itself by setting the prices even if it were to know the actual net benefit functions of the users. Namely, there is a function $\Delta(\epsilon)$, with $\Delta(\epsilon) \to 0$ as $\epsilon \to 0$, such that if the network's estimates

[4]For the modified Jacobi scheme, we need to assume that everyone sends a positive flow at the NEP.

of the actual net benefit functions are ϵ-close according to (4), then the overall benefit to the network is within $\Delta(\epsilon)$ of what it would achieve even if it knew the actual net benefit functions of the users. This is formalized by the following lemma.

Lemma 2 *Let the number of servers be $L \geq 1$. For all $\epsilon > 0$, there exists $\delta(\epsilon) > 0$ such that $\delta(\epsilon) \to 0$ as $\epsilon \to 0$ and such that*

$$\max_{i,j} |\tilde{r}_l^i - r_l^{i*}| < \delta(\epsilon)$$

if $d(U^i(\cdot), J^i(\cdot)) < \epsilon$ for all $i \in I$, where $\tilde{r}$ is the actual NEP and r^* is the NEP the network expects based on its estimated net benefit functions $J^i(\cdot), i \in I$.

Lemma 2 tells us that if the network can obtain good approximates for the net benefit functions of the user, using the form in (2), the network can use a pricing mechanism as an effective means of controlling the network loads.

7 Numerical Example

In this section we give an example with two servers and five user. Although the theorems proved in section 5 apply only to the case with a single server, if the interdependency between the flow rates to different servers is not too large, i.e., $K_i, i \in I$, are sufficiently small, the flow rates are expected to converge even in multiple server cases. The service rates of servers 1 and 2 in this example are 8.0 and 6.5, respectively. We assume that the network already has the correct values of the parameters in the benefit functions of the users.. These are given in Table 1. The objective of the network is to keep the load to the servers 1 and 2 at 5.0 and 4.0, respectively. Suppose that the network wants the users to send the data at the rates given in Table 1. We assume that $K_i = 0.02$ for all $i \in I$, and using the rates in Table 1, compute the price vector that should be used by the network. These prices are also given in Table 1.

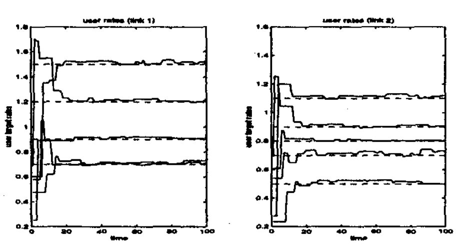

Figure 2: Convergence to NEP under Gauss-Seidel update scheme.

The initial rate of a user for each server is randomly selected from [0.2 0.7], and Figures 2 and 3 show the convergence of user flow configurations to the unique NEP. Since users may not be able to measure the rates of other users correctly, we have allowed the measurement noises of up to five percent. These figures clearly demonstrate that the users do converge to the NEP. Similar convergence results have been obtained from simulations, with two servers and up to 55 users.

User	α	β	Server	Rate	price
1	0.5	0.5	1	1.20	0.390
			2	0.9	0.449
2	0.7	1.0	1	1.5	0.443
			2	0.7	1.081
3	0.6	0.5	1	0.9	0.745
			2	0.8	0.693
4	0.8	0.7	1	0.7	1.427
			2	0.5	1.392
5	0.4	0.6	1	0.7	0.551
			2	1.1	0.151

Table 1: The parameters of the users, the desired rates from the users and the corresponding prices.

References

[1] D. Fudenberg and J. Tirole. *Game theory.* MIT Press, Cambridge, MA, 1991

[2] R. J. La and V. Anantharam. "Network pricing using game theoretic approach" - full version, available at *http://www.path.berkeley.edu/˜hyongla*

[3] A. Lazar, A. Orda, and D. Pendarakis, "Virtual path bandwidth allocation in multi-user networks", *Proc. IEEE INFOCOM '95*, Vol. 1. pp. 312-20, 1995.

[4] D. G. Luenberger. *Introduction to linear and nonlinear programming.* Addison-Wesley Publishing Company, Menlo Park, CA, 1973

[5] J. Rosen, "Existence and Uniqueness of Equilibrium Points for Concave n-person games", *Econometrica*, Vol. 33, pp. 520 -524, July 1965.

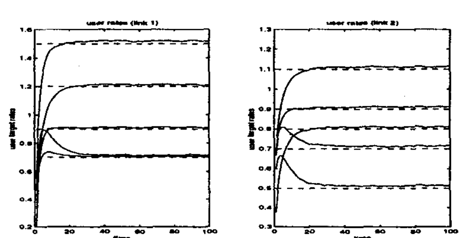

Figure 3: Convergence to NEP under Jacobi update scheme.