Chapter 17

PRICING VIRTUAL PRIVATE NETWORKS - AN ECONOMIC, ENGINEERING AND EXPERIMENTAL APPROACH

Zhangxi Lin[*], Peng Si Ow[+], and Dale O. Stahl[**]

Keywords: Network traffic pricing, virtual private networks, queueing model, agent-based simulation, direct experiment.

Abstract: This chapter presents a network traffic-pricing model for virtual private network (VPN) deployed on packet-switching networks. A transaction-level pricing architecture based on proxy server technology is proposed for the implementation. Analytical expressions of pricing formulas for first-in-first-out and round-robin bandwidth are derived. Both agent-based simulations and the human subject based direct experiment have been conducted using real-time test data. The experimental outcomes strongly support that the pricing mechanism can effectively improve a VPN's transmission efficiency measured by the service welfare rate.

1. VIRTUAL PRIVATE NETWORKS AND THEIR TRAFFIC MANAGEMENT PROBLEM

The virtual private network (VPN) (Kosiur, 1998) is a value-added network built upon various types of network clouds, particularly on the Internet, with secured virtual paths for data transmission. The explosive

[*] College of Business Administration, Texas Tech University, *zlin@ba.ttu.edu*

[+] Center for Research in Electronic Commerce (CREC), The University of Texas at Austin, phighnam@aol.com

[**] Department of Economics, The University of Texas at Austin, *stahl@eco.utexas.edu*

growth of the Internet is also spilling over to Internet-based VPNs, so that VPNs are emerging as an important enterprise networking solution for corporations. The cost-effectiveness of Internet-based VPNs has stimulated the demand for VPN services. According to Infonetics Research, worldwide expenditures on VPNs should double annually through 2002 when they are expected to reach $20 billion. Yankee Group predicts that by the year 2003, VPNs will be used by 70 percent of all companies for up to 90 percent of their data communications needs.

Although the Internet-based VPN can reduce the monetary cost of network applications, Internet traffic congestion also reduces the benefit from the use of VPNs as a result of exponentially increasing traffic loads. In its "*Top 10 Discoveries About the Internet*" [http://www.keynote.com/measures/top10.html], Keynote Systems listed the Internet performance problem as the first among the ten. In one of its weekly reports, Keynote System's *Keynote Business 40 Internet Performance Index* showed that the best response time was about 1.5 seconds in a web site access and the worst average was about 15 seconds.

The history of Internet is in fact a recursive process of performance improvement and demand growth. The high demand urges better Internet service quality and the improved Internet services results in higher demand. In late 1980s, a group of flow-control algorithms such as *slow-start* and *congestion avoidance* were proposed and widely implemented today (Jacobson and Karels, 1988; Nagle, 1984; Stevens, 1997), which made Internet services stable from the collapse caused by network congestion. In addition, queue management algorithms for Internet transmission nodes have been designed to complementarily allocate bandwidth and deal with queue overflow for network nodes along data paths (Braden et al, 1998). Recent research in active queue management algorithms, such as *Random Early Detection* (RED) (Floyd and Jacobson, 1993) and *fair queuing* (FQ) (Demers, Keshav and Shenker, 1990), has led to more powerful networking products that can provide better quality of service (QoS).

The major idea behind these technical schemes to improve Internet services is feedback. With feedback in a flow control session, the sender and the receiver computers of an end-to-end connection dynamically exchange the information about available bandwidth and maintain an appropriate transmission bandwidth. Furthermore, heavily loaded nodes can send back alerts to the origins of data flows to trigger responses to the congestion. If the network communication software for the sender is responsive to these mechanisms, it will automatically reduce the data rate to avoid the congestion.

However, there are two problems with these non-incentive-compatible approaches. First, they are effective only if the network applications are responsive. Those data flows generated by "non-responsive" applications can get around the flow-control mechanism to obtain more bandwidth (Floyd and Fall, 1999), and therefore deteriorate the network performance. Because of the ever-increasing heterogeneity of Internet protocols, many of which do not comply with the traffic control algorithms, these traditional approaches are no longer working properly. Second, a pure technical scheme is unable to discriminate among different types of data flows in accordance with their values. The only constraint to the overexploitation of the network bandwidth resource is the throughput time when the network is overloaded. In this case, the data flows sensitive to the delay are affected more regardless of their value to users. Presently, there is no final solution for the Internet congestion problem. Hence traffic congestion control is still an important research topic (Allman, Paxson and Stevens, 1999).

Since the 1990's, there is an emerging consensus that the Internet traffic congestion problem is not merely an engineering issue, but a problem of allocating scarce network resources to users whose valuations of these resources vary. Increasingly research has been conducted on economic network resource allocation mechanisms that support usage-based pricing and incentive compatibility (Clark, 1996; Gupta, Stahl and Whinston, 1999). Examples are *dynamic bidding for access* by MacKie-Mason and Varian (1995); *priority pricing* by Gupta, Stahl and Whinston (denoted as the GSW model) (Gupta et al, 1997; Gupta, Stahl and Whinston, 1997; Li et al 2000); *edge pricing* by Shenker et al (1996); *Paris metro pricing* by Odlyzko (1997); and *progressive second price auction* by Lazar and Semret (1998).

Although researchers theoretically proved that the economic approach has the potential to solve Internet congestion problems, they also realized that a good traffic-pricing model must come with an implementation scheme that is proved to be practical. It must be convincing to computer scientists that a mathematically intensified economic approach would not disturb the operation of the current mechanism for network traffic management and congestion control, and would work well alongside the existing technologies.[1]

Our research in VPN traffic pricing is focused on the implementation feasibility of the network traffic-pricing scheme. The reasons that the Internet-based VPN is chosen as the target network are:

- The VPN's performance has become a significant issue in its business applications because the encrypted packets impose more traffic loads which reduce the benefits.
- The VPN possesses some useful business features such as user authentication and user account management that make the implementation of traffic-pricing feasible.[2]

The model developed in this research follows the methodology used by the GSW model, which was initialized by Stahl and Whinston (1991) and enriched by Gupta, Stahl and Whinston (1997). The GSW model is a general equilibrium model with a resource-price structure that is incentive compatible for network resource allocation. It was tested under various scenarios by a simulation of a public network. The simulation indicated that traffic pricing can significantly improve network service benefits and the service prioritization will lead to better outcomes. In theoretical aspect, we have enhanced the GSW model to fit packet-switching networks that use either first-in-first-out (FIFO) or round-robin (RR) bandwidth scheduling. In practical aspect, we have developed the architecture and technology for a prototype of VPN traffic-pricing system. The experimental outcomes from the prototype system have conversely verified the correctness of the theoretical results.

2. A TRANSACTION-LEVEL TRAFFIC-PRICING SCHEME

Before a dynamic traffic-pricing model can be developed, we need to know how it is used and where it is enacted. In particular, the topology and data flow control scheme for a VPN may critically affect the form of the pricing formula. A typical Internet-based VPN can be built up with special network devices to connect geographically distributed LANs into a virtual intranet/extranet over the Internet. The constructive hardware includes firewalls, certificate authority servers, security gateways, etc., underpinned by the security technologies such as security transmission protocols (e.g. IPSec), user-authentication information management protocols (e.g. LDAP), key-management protocols (e.g. ISAKMP), etc. Logically, we refer to the hardware set supporting the VPN technology as the *VPN gateway*. Therefore, a VPN gateway is an enhanced network gateway between an application domain and the Internet. It can provide the required security functions such as:

- Wide-area network tunneling, i.e., establishing a secure network connection across the public Internet;
- Data encryption;
- Filtering/firewalling, i.e., security control of incoming and outgoing packets at network edges; and
- User authentication.

The application of encryption/decryption and related security protocols has added more traffic and processing overhead to VPNs and is blamed for the worsened congestion problem. Here, we define network traffic congestion as prolonged time for a data flow to get through the network compared with the time it would take to get through an idle network[3]. By this definition, the congestion may result from the queue waiting time at network nodes as well as from queue overflow.

We propose a *transaction-level pricing* architecture for the VPN traffic pricing to solve implementability and efficiency problems. We claim that it allows us to solve relevant issues for a network traffic-pricing system, such as: digital contracting; pricing system efficiency; logistic system operation (payments, accounting, etc.) irregularities; problems involved with the integration of the traffic-pricing system and existing traffic-control techniques; and user acceptability. Proxy server-based VPN traffic pricing is the underpinning infrastructure for transaction-level VPN pricing architecture (Figure 1). A proxy server is employed as the bandwidth broker to schedule the data flows with a pricing mechanism for an affiliated VPN gateway, which is called *traffic proxy server* (TPS). TPSs can be deployed somewhere between VPN security devices and application domains, for example, between VPN gateways and local area networks.

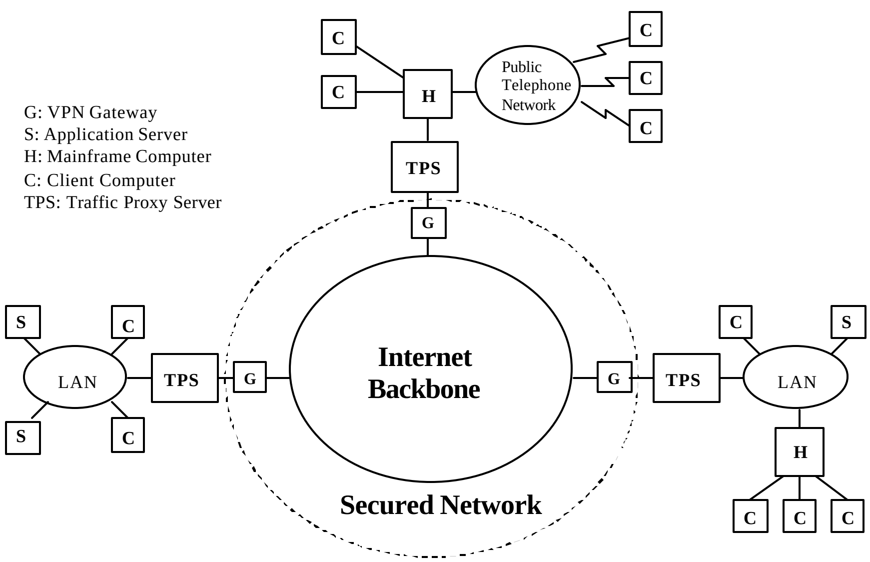


Figure 1. The infrastructure of proxy server-based VPN traffic-pricing system

The terms used in this chapter include: A *Job* [4] is a synonym for *Transaction*[5] and is defined as a unit of a network service requested by a user, which may generate one or several data flows transmitted through a network. A *data flow* is a group of IP packets controlled by, for example, a TCP connection. *Job size* is referred to as the total volume of data flows incurred by a job. It is measured in the number of segments composed of several packets in the same data flow. *Throughput time* is used to describe the transmission time for a group of packets, e.g. a data flow, going through a section of transmission channel. The throughput time of a data flow includes its transmission time and waiting time at a given channel. It is determined by the traffic load conditions as well as routing and transmitting disciplines. Transaction-level pricing is to be implemented on top of transport layer protocols without looking into the internal mechanism at the lower layer for data flow delivery and control. With the transaction-level implementation, VPN gateways can schedule data transmission tasks in regard to the application needs and priorities.

We suggest a two-session interactive model for TPS operation. A similar model has been designed in practice for secure proxy servers (Oppliger, 1998). One of the sessions is designed for security control purposes and another for data transmission service. The following process provides an example for a data retrieval transaction:

- A remote client on a user's behalf requests a connection to the TPS with the user's ID and password.
- If the user's information is properly authenticated, the client is permitted to proceed to request information services from the application server that the TPS proxies. Meanwhile the TPS checks an active user list. If the user/client pair is not on the list, the TPS adds the user/client to the list. The TPS periodically sends pricing information to all active clients according to the active client list, so that users are able to make job submission decisions.
- The user may submit jobs after the primary control connection is properly processed. Only the jobs with positive net values are submitted, which are calculated by a smart agent using the expected throughput time and the price that the client received from the TPS through the primary connection. In this stage, the client establishes a secondary connection to the TPS.
- The TPS authenticates and authorizes each job request and relays it to a destination application server.
- The application server completes services and sends back flows of data to the TPS.
- The TPS schedules data transmission tasks for the affiliated VPN gateway and bills the services to the user's account according to the jobs' sizes and the user's QoS requisitions.
- The TPS forwards data flows to the client.

The major difference between the interaction model suggested for the TPS and the one used for the secure proxy server is that the TPS schedules data flows and accesses user accounts each time that an application server generates data flows, while the SPS function is accessed before a service starts. Some protocols, such as *Remote Authentication Dial-In User Service*

(RADIUS) that has been used by some VPN solutions (Rigney and Livingston, 1999; Rigney et al, 2000), can be utilized for the TPS in user authentication and remote network connection administration.

3. A DYNAMIC TRAFFIC PRICING MODEL FOR VPNS

Several restrictions are necessary to narrow down the VPN traffic-pricing problem. First, we focus our discussion on the data flows incurred by application service requests but neglect the traffic for the routing and addressing services, such as domain name service and routing information distribution, because they have relatively less impact on the network bandwidth. Second, in the main part of this paper we will discuss the traffic-pricing model based on the assumption that a job incurs a single data flow. We can extend the result to a general situation, where one job is related to multiple data flows[6]. This strategy allows us to simplify the model derivation without losing the correctness of the outcome. Third, we do not consider traffic management in the backbone because it is out of the organization's control. We assume that Internet service providers assure the requested QoS on the Internet in accordance with the contracts, such as a service-level agreement. Finally, traffic control within LANs is not considered. This is because LANs normally have enough bandwidth. Therefore, we can concentrate on LANs' Internet connections, which could be the network bottlenecks because their bandwidth is restricted by an organization's networking budget.

3.1 The Characteristics of VPN Queueing Model

An Internet-based VPN can be defined as a set of $G = \{g_s\}$ gateways, $s \in S$, an equal number of channels $C = \{c_s\}$ connecting gateways to the Internet, providing that each gateway has only one Internet connection, and Internet tunnels that are dynamically established for secured transmissions. With the assumption of the assured QoS of Internet tunnels, the bottleneck of a VPN route, if it exists, is one of the two channels between VPN gateways and the Internet, not the Internet tunnel. Consider a route R^j carrying data flow j from a sender to a receiver. R^j is a set of nodes and channels. It can be denoted as $R^j = \{g_1^j, g_2^j, c_1^j, c_2^j, c_n\}$, where g_1^j is the gateway with a bottleneck Internet connection c_1^j, g_2^j is the gateway with a non-bottleneck Internet connection c_2^j, c_n^j is an Internet tunnel for data flow j. g_1^j and $g_2^j \in G$, c_1^j and $c_2^j \in C$. Here, we say channel c_s^j is a bottleneck in route R^j referring to a data flow j if the channel's cumulative available bandwidth capacity is always less than another channel's cumulative available bandwidth capacity in the time period servicing the data flow. If the data transmission request at a VPN gateway is a Poisson process with a general size distribution, the bandwidth allocation service becomes an M/G/1 queueing system because the capacity of a channel is deterministic but the size of the data flows varies (Kleinrock, 1975).

Generally, the above VPN route can be modeled as a three-stage queueing system (Figure 2): the first queueing server is the Internet connection channel c_1 for outgoing data flows from the sender; the second one is the Internet tunnel c_n; and the third one is the Internet connection channel c_2 for incoming data flows to the receiver. The traffic loads on channel c_1 also come from other computers in the LAN the sender locates. A major portion of outgoing data flows from channel c_1 is sent to other destinations than the receiver being discussed, and therefore they will not join the queue for channel c_2. In a similar way channel c_2 also transmits data flows from other sources to the LAN where the receiver locates without increasing the burden on channel c_1.

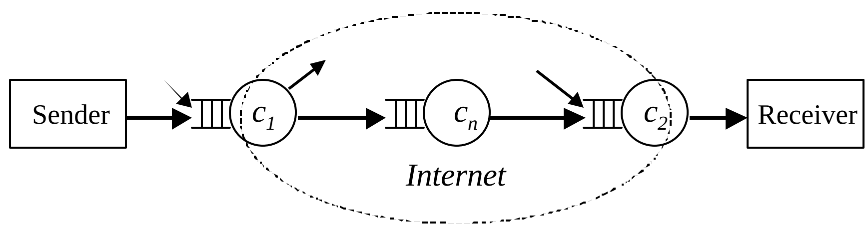


Figure 2. A three-stage queueing model for a VPN route.

Even though the capacity of the Internet connection channels is the main concern in traffic control, the channels are passive to traffic loads, and routers/gateways are actually allocating bandwidth for them. The traffic through channel c_1 can be observed and measured at gateway G_1 that maintains the queue for channel c_1 and allocates the bandwidth for c_1. Although the queue for channel c_2 forms at a router in the Internet, the traffic through channel c_2 can still be observed and controlled at gateway G_2. In this context, the VPN gateway and the transmission channel are equivalent in traffic control. If an Internet connection channel is the bottleneck in a route, we can also refer to the affiliated VPN gateway as the bottleneck gateway.

One of the important advantages of packet-switched networks is that the total delay a data flow encounters in a route is less than the summation of waiting and servicing times in all channels (or equivalently nodes) along the route. This feature allows packet-switched networks to provide better bandwidth efficiency than other types of network, such as circuit-switched networks. However, the "pipeline" effect in a packet-switched network adds complexity in setting up the price for traffic because the total throughput time is not a sum of the throughput times at each individual packet-forwarding device. The same effect is applied to VPNs built on the Internet.

Proposition:

In a packet-switched network, the total throughput time that a data flow is transmitted through a route can be expressed as the time spent at the bottleneck node plus a relatively trivial amount of delay on other nodes of the path.[7]

The following is an intuitive explanation of the proposition:

When a data flow is transmitted through a route in a packet-switched network, packets are sent just like water flows through a pipe. Every node in the route forwards the packets through the channel to next node whenever it receives the packets and the channel has available bandwidth. Because of time overlapping, the total throughput time for the data flow is less than the summation of throughput times at each node. By intuition, the time that the data flow passes through the bottleneck node will almost overlay the throughput time at any other node. There is only a minor difference in transporting a unit of the data flow, which has been defined as a segment of packets. The number of packets in a segment depends on the flow control algorithm adopted by the network. Therefore, the total time of the data flow transmitted through the route can be expressed as the throughput time for the bottle node plus the time for a unit of packets to be sent through the route.

Applying the above proposition to the VPN, we can infer that the total transmission time for a data flow through a VPN route is the throughput time of the bottleneck Internet connection plus the time for a segment of packets, which is the processing unit of packets, to go through the other path.

3.2 VPN Traffic-Pricing Formulas

The VPN traffic-pricing model is assumed to comply with the following economic conditions:

- A job has an intrinsic value perceived by the user who generates it;
- A job's net value to the user depends on three factors: the intrinsic value, the price charged, and the delay cost which is proportional to the throughput time;
- Prices are set for the channel bandwidth consumption in accordance with the network traffic status;
- The information of both the bandwidth price and the expected throughput time are periodically disseminated to users;
- Users are rational, i.e., they submit their jobs only if expected net gains from job submissions are positive.

The protocol for VPN traffic pricing is a three-stage process:

1) A TPS periodically decides the bandwidth usage price, based on its current traffic load status, for the bandwidth service between the local network and the Internet. The price and the traffic load status are disseminated to those client computers that are using the VPN transmission services.

2) A smart agent, i.e., a client side application, helps users estimate jobs' values and sizes, as well as the effects of delay on the jobs. It automatically makes job submission decisions according to the net values of the jobs by taking account of transmission service prices and throughput times. These decisions are imprecise, but statistically the errors are acceptable because users can update the rules for job size predictions and job value estimations, and therefore improve the smart agent's performance.

3) Once a job is submitted it will be charged with the price by the real size of total data flows serviced by the VPN. The costs are billed to the user's account.

In the transaction-level pricing architecture, the TPS replaces the VPN gateway to allocate the bandwidth of Internet connections. We investigate three bandwidth-scheduling algorithms, prioritized FIFO bandwidth scheduling, non-prioritized RR bandwidth scheduling, and prioritized RR bandwidth scheduling.

The priority is one of the user’s choices in data transmission services, which makes a difference to the delay cost. In prioritized FIFO scheduling, if the queue of a higher priority is nonempty the service for the queue of a lower priority will never begin. In another aspect, we assume there is no preemption once a data flow transmission has begun. That is, if a data flow at a lower priority is being serviced and another data flow at a higher priority arrives, the data flow at the lower priority is continued until completion.

The prioritized RR bandwidth scheduling adopts a preemptive policy. That is, a data flow at higher priority obtains bandwidth immediately after it arrives the server regardless the service status of jobs at the lower priority classes. A new data flow joins the tail of the queue for the requested priority class and is assigned a fixed-length time slice in its turn if there is no data flow in higher priority queues. With prioritized RR bandwidth scheduling, a VPN gateway allows all data flows being serviced in the same priority class to share Internet connection channels equally. The non-prioritized RR is the special case when there is only a single priority class.

From the user’s viewpoint, we define *job type* to distinguish the characteristics of a job, such as job size and application type. Different users may submit the same types of jobs at different moments. However, jobs of the same type remain identical in size and application type, even though their values vary from user to user and from time to time. We model the demand for transmission services as determined by (i) an exogenous “potential” rate which would prevail if all costs were zero, and (ii) the monetary and time costs of obtaining the service. Let *I* denote the set of users, *J* the set of job types, and *Q* the set of job sizes. Let l_{ij} be the exogenous potential service request rate of job type *j* from the user *i*. The total exogenous rate, i.e., the maximum potential job rate, for the VPN is $l = \sum_{i=1}^{I} \sum_{j=1}^{J} l_{ij}$, and the maximum potential data flow rate is $\sum_{j=1}^{J} q_j \sum_{i=1}^{I} l_{ij}$, where q_j $\hat{I}$ Q is the size of a type *j* job measured in the number of segments. These two forms of exogenous rates are based on the aggregation of the demands when there are no costs—no transmission charges and no response delays that impose extra costs to services. A cost reduces a service’s net value and hence prevents the submission of lower-valued jobs. Therefore, in reality, not every potential service request will be submitted because the costs from service delays are inevitable. The real demands in responding to the price and the delay time are the rates of jobs submitted to the network.

The above conditions allow us to setup a benefit maximization problem for VPN bandwidth service (Lin et al, 2000; Lin, Stahl and Whinston, 2000). By solving a VPN benefit-maximization problem, we can obtain a general expression of VPN bandwidth rental price for a given channel:[8]

$r_{qk}^{*} = \sum_{l \in I} \sum_{m \in J} \sum_{h \in K} \frac{\partial \Omega_{mh}}{\partial j_{qk}} x_{lmh} d_{lm}$, if the channel is the bottleneck of route R^j; *(3.1a)*

$r_{qk}^{'} = \sum_{l \in I} \sum_{m \in J} \sum_{h \in K} \frac{1}{q} \frac{\partial \Omega_{mh}}{\partial j_{qk}} x_{lmh} d_{lm}$, if the channel is **not** the bottleneck of route R^j. *(3.1b)*

where W_{mh} is the expected throughput time—the time that a job gets through a VPN gateway—for type-*m* job submitted to priority-*h* service, x_{lmh} is data flow rate of type-*m* job submitted by user *l* for priority-*h* service, d_{lm} is delay cost coefficient for type-*m* job submitted by user *l* indicating the impact of delay on user’s benefit, and j_{qk} is job rate for jobs of size *q* submitted to priority-*k* service. Size *q* is defined as $q = q_j$ for type-*j* jobs.

Generally, $q >> 1$, hence $r_{qk}^{*} >> r_{qk}^{'}$. Then we can approximately apply a single price r_{qk}^{*} to type-*j* job submitted by user *i* for service priority *k* neglecting the effect of the non-bottleneck price $r_{qk}^{'}$.

The total price paid for a job *j* at priority *k* of the channel can be expressed as:

$$r_{jk} \text{ » } r^{*}_{qk} = \sum_{l\in I} \sum_{m\in J} \sum_{h\in K} \frac{\partial \Omega_{mh}}{\partial j_{qk}} x_{lmh} d_{lm} \qquad (3.2)$$

With FIFO scheduling, the expected throughput time depends on the expected waiting time in the queue, w_k, which is invariant to a job's type, and the channel service time, which is invariant to priority class once the job is being serviced. Therefore, the queue waiting time w_k for a M/G/1 queueing system can be expressed in terms of job size q instead of job type j:

$$w_k = \frac{\sum_{h\in K} \sum_{q\in Q} j_{qh} q^2}{2B^2 (1-\sum_{h<k} r_h)(1-\sum_{h\leq k} r_h)} \qquad (3.3)$$

Then the throughput time:

$$W_{jk} = w_k + q / B \qquad (3.4)$$

where B is bandwidth.

By using the above formula, we can obtain a pricing formula for a FIFO scheduling that is quadratic in job size:

$$r^{*}_{qk} = \sum_{h\in K} j_h \bar{d}_h (a_{1h} q + a_{2h} q^2) \quad "q \text{ Î } Q, k \text{ Î } K \qquad (3.5)$$

where $a_{1h} = \dfrac{w_h (2-\sum_{l<h} r_l - \sum_{l\leq h} r_l)}{B\ (1-\sum_{l<h} r_l)(1-\sum_{l\leq h} r_l)}$ when $k < h$, $a_{1h} = \dfrac{w_h}{B\ (1-\sum_{l\leq h} r_l)}$ when $k = h$,

$$a_{2h} = \frac{1}{2B^2 (1-\sum_{l<h} r_l)(1-\sum_{l\leq h} r_l)},$$

r_l is the bandwidth utilization ratio for priority l service, $\bar{d}_h$ = $\sum_{l\in I} \sum_{m\in J} \frac{x_{lmh}}{j_h} d_{lm}$ is the mean of d_{lm} over user l and job type m, weighted by the flow in priority h, and $j_h = S_q\ j_{qh}$ is job arrival rate at priority-h queue.

In a round-robin scheduling system, a job's throughput time is proportional to job size and the average number of jobs in the queue during servicing. The throughput time of a size q_j job in a non-prioritized queue is $t_j = t(q_j) = [(L^*+1)\ q_j - r/2]/B$, where $L^* = \dfrac{r^2 E[q^2]}{(1-r)(E[q]+E[q^2])}$ is the average number of jobs being serviced, r is the bandwidth utilization ratio of the gateway, and $E[q]$ and $E[q^2]$ are the expected size and size-squared values for the gateway. We can derive the optimal unit price (Lin et al, 1999):

$$r_0 \text{ » } \frac{\bar{d}}{B}(L^* + \frac{L^* - r}{1 - r}) , \qquad (3.6)$$

where $\bar{d} = \sum_{l \in I} \sum_{m \in J} x_{lm} d_{lm} q_m / \sum_{l \in I} \sum_{m \in J} x_{lm} q_m$ is the mean of d_{ij} over i and j, weighted by data volume rates. The approximate form for the pricing with RR scheduling indicates that the expected number of jobs in the queue is a critical factor in a job's price and the price is proportional to the size of a job.

In a prioritized RR bandwidth scheduling, the price for type-*j* job submitted to class-*k* is (see the Appendix in Lin, Stahl and Whinston, 2000):

$$r_{jk} = \sum_{m \in J} \sum_{h > k} j_{mh} \bar{d}_m \frac{q_{mh} q_{jk} L_h{}^2 [2(1 - \sum_{l<h} r_l) - r_h]}{B^2 (1 - \sum_{l<h} r_l)^2 r_h}$$

$$+ \sum_{m \in J} j_{mh} \bar{d}_m \frac{q_{mk} q_{jk} L_k{}^2}{B^2 r_k{}^2}$$

$$= \frac{q_{jk} L_k{}^2}{B^2 r_k{}^2} \frac{2(1 - \sum_{l<k} r_l) r_k - r_k{}^2 + (1 - \sum_{l<k} r_l)^2}{(1 - \sum_{l<k} r_l)^2} \sum_{m \in J} \sum_{h > k} q_{mh} j_{mh} \bar{d}_m \qquad (3.7)$$

The unit price for priority k service is:

$$r_k = \frac{L_k{}^2}{B^2 r_k{}^2} \frac{2(1 - \sum_{l<k} r_l) r_k - r_k{}^2 + (1 - \sum_{l<k} r_l)^2}{(1 - \sum_{l<k} r_l)^2} \sum_{m \in J} \sum_{h > k} q_{mh} j_{mh} \bar{d}_m$$

$$= \frac{L_k{}^2}{B r_k{}^2} \frac{(1 - \sum_{l \le k} r_l)^2 - 2 r_k{}^2}{(1 - \sum_{l<k} r_l)^2} r \bar{d} \qquad (3.8)$$

where $r = \sum_{k \in K} r_k$, $\bar{d}$ is the flow rate weighted delay cost coefficient.

4. TRAFFIC PRICING EXPERIMENTS

4.1 Experiment Design

We developed a prototype system named *VPN Traffic-Pricing Experiment System* (VTPES) (Lin et al, 1999) to test the transaction-level pricing architecture and to conduct experiments for the pricing model. VTPES is built on a small network platform in the Center for Research in Electronic Commerce (CREC) at UT Austin. Currently, it has six major distinctions from the previous GSW model simulation system:

1) It is a real-time system. Both data traffic generation and bandwidth allocation are implemented on a real-time basis.

2) It is a scalable distributed system. VTPES runs with a real network consisting of several computers. The configuration of the network can be varied to test the performance of VPN traffic pricing under certain definable conditions.

3) It has a dual-queue structure. In addition to a regular queue structure for bandwidth services, an extra queue is configured as a benchmark system. By using a shared traffic generation source, we can test different experimental schemes and compare the outcomes with that of a standardized scheme.

4) It can carry out both agent-based simulations (Tesfatsion, 2000) and human subject based direct experiments.

5) It can record data flow patterns generated in a direct experiment and replay the data flow patterns later using agent-based configuration. In this way, assessing a traffic-pricing scheme's performance in a real environment becomes possible.

6) Only non-priority bandwidth scheduling is implemented that already satisfies the purposes of the experiment.

Implementation of the experimental system that allows comparison of performance between human subjects and computer agents provides a good means for us to move the research closer to a real world. It becomes possible to bridge the gap between the economic approach and behavioral approach particularly in the network traffic-pricing area (Sterman, 1987). Logically, VTPES consists of four modules (Figure 3):

1) A virtual bandwidth server allocating bandwidth with a round-robin scheduling method for service requests;

2) A set of user-oriented applications operating on the web browser to generate service requests and display network status for request submission decisions;

3) A web-based application module as an intermediary between users and the virtual bandwidth server; and

4) A traffic load generator, which is an agent for generating or regenerating network data flows.

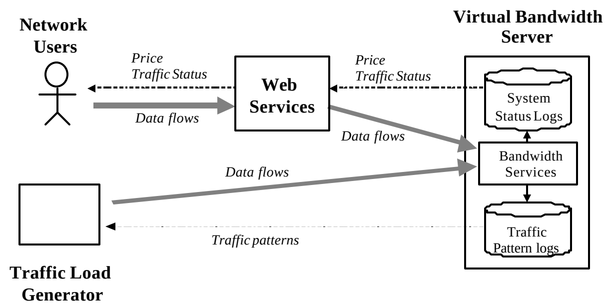


Figure 3. Current VTPES Logical Structure

VTPES can operate in three operation modes: agent-based, direct experiment, and mixed, i.e. computer agents and human subjects work together. When working in the agent-based simulation mode, VTPES can be configured with five components representing a network route: a client computer, a client-side VPN gateway, a server-side VPN gateway, a TPS, and an application server (Figure 4). The client computer generates jobs and submits those jobs that are expected to create net benefits after taking out all costs. The client-side VPN gateway routes the jobs to the destination application server via the server-side VPN gateway. The application server services the jobs relayed by the TPS and generates data flows back to the client computer. The TPS performs pricing, bandwidth scheduling and user-access accounting, etc.

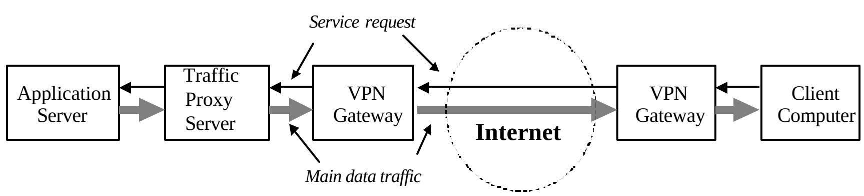


Figure 4. A unit VPN traffic-pricing system

This 5-tier pricing system can be reduced to a smaller scale to ease the experiment without losing the desired experimental features. The client-side gateway is “transparent” to data flows and can be ignored. The server-side VPN gateway’s bandwidth scheduling function can be merged to the TPS because the TPS and the gateway can be logically considered as an integrated subsystem. Thus, the server-side gateway can also be ignored, handing over its functions to the TPS. In fact, the terms, VPN gateway and TPS, are exchangeable when referring to the experiment components.

4.2 Pricing Effectiveness

We mainly examined the welfare rate, i.e., per-second service welfare that is the output being monitored, to evaluate the effect of different scheduling schemes for non-priced and priced systems on VPN bandwidth service performance. The network load parameters for the experiment, i.e., the input, include the distributions of job size, the job value, and the delay cost coefficient. The job value and delay cost are random variables depending on the user's timely preferences and can be generated by the computer. Job size can be predefined in a profile.

In order to test whether the performance from the experiment based on a real network is the same as that from the GSW software-based simulation, we intensively used the same parameter values for most experiment schemes as those used in the GSW simulation. Although we also designed experiments with different job parameter distributions, using GSW experiment parameter values allows us to compare the data obtained from our experimental schemes with the ones from GSW model experiments.

The job value distribution used in the GSW model simulation has a normal distribution with a mean of 500 dollars and a standard deviation of 150 dollars, i.e., job value ~ $N [500, 150^2]$. It is independent of job size. The delay cost coefficient distribution is also independent of job size. The GSW model uses a normal distribution for the delay cost coefficient with the mean of 4 dollars and the standard deviation of 1 dollar, i.e., job delay cost ~ $N [4, 1]$. The absolute levels of the job value and the delay cost coefficient are not important, but the relative levels and their correlation to job size are critical. Therefore, in addition to this basic set of distributions, we also tested two diversities: a set of job parameters with different standard deviations, and a set of the parameters that are relevant to the job size.

We conducted this group of experiments in four steps:

1) Verify the consistency of the outcome from a network-based experiment and the one from the GSW model simulation using FIFO scheduling;
2) Test the performance of the VPN using non-priced RR scheduling;
3) Compare the outcomes of priced schemes between RR scheduling and FIFO scheduling; and
4) Test the priced schemes using different job parameters.

The outcome from the simulation running on VTPES using the FIFO scheduling matches that from the GSW model simulation very well. VPN traffic pricing significantly improves network welfare rate in FIFO scheduling (Figure 5a). The curve of the welfare rate from the non-priced FIFO scheme starts to decline when exogenous traffic rate increases and approaches capacity, while that from the priced FIFO scheme keeps going up as the traffic rate increases. This is exactly the same result as the GSW model simulation revealed.

The experiment shows that priced FIFO scheduling performs better than non-priced RR scheduling (Figure 5b). The welfare rate yielded from a non-priced RR scheme increases with the augmented traffic rate. This is because the expected throughput time for a job in RR scheduling is proportional to the job's size. This provides the incentive for users not to submit the jobs with lower unit values. Since the submission decision is based on the comparison between the unit job value and the unit waiting cost, the efficiency of the network is better than that of non-priced FIFO. However, the increase rate of the welfare rate from a non-priced RR scheduling scheme is much lower that that from a priced FIFO scheme. The gap widens when the traffic rate goes up.

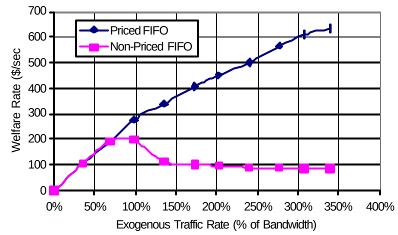
The experiment demonstrates that pricing is also effective in the RR scheduling scheme with approximately the same welfare rate as that from priced FIFO scheduling (Figure 5c). Pricing on FIFO and RR scheduling systems results in almost the same job submission ratio distributions (Figure 5d). In the chart, the *x*-axis is the job size index with larger numbers for larger job sizes and the *y*-axis is the percentage of jobs having been actually submitted in terms of an exogenous job rate. Both the submission ratio distributions consistently drop when the job size is getting larger.

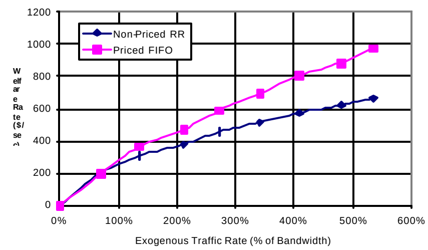
Although it is necessary to make the outcome of VTPES-based experiments comparable to the previous results from the GSW model, a thorough and complete experimentation should cover more variable factors to provide strong evidence in supporting our conclusions. There are two versions of schemes in this step. The first experiment version uses the parameters with new job value distribution $\sim N\ [500, 200^2]$ and job delay cost coefficient $\sim N\ [4, 2^2]$. As the chart in Figure 6 shows, the welfare rate obtained from the scheme using the new job value and delay cost distribution has only a negligible difference from that using the parameter values designed for the GSW model simulation.

The second experiment version uses a variable delay cost distribution and job value distribution that are relevant to the job size. We use the following conversion formula to make delay cost or job value relevant to job size:

$$X^\wedge = X * (J / B)^{s*0.5}$$

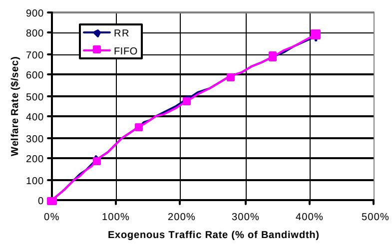
where J is job size, B is bandwidth, $s = 1$ when a job value is converted and $s = -1$ when a delay cost coefficient is converted, X is a regular value of the delay cost coefficient or a regular job value, and $X^\wedge$ is the converted value. The rationale behind the above conversion formula is that the larger a job's size is, the higher the job's gross value could be or the lower the delay cost coefficient could be.

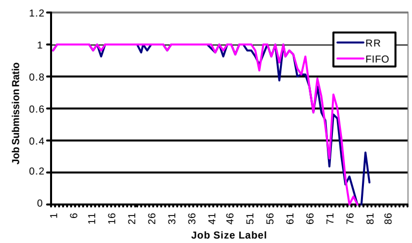




(a) Priced FIFO versus non-priced FIFO

(b) Non-priced RR versus priced FIFO





(c) Welfare rate comparison between priced RR scheduling and priced FIFO scheduling

(d) Job arrival distribution comparison between priced RR and priced FIFO schemes (exogenous traffic rate: 4 Mbps)

Figure 5. Effectiveness of VPN traffic pricing

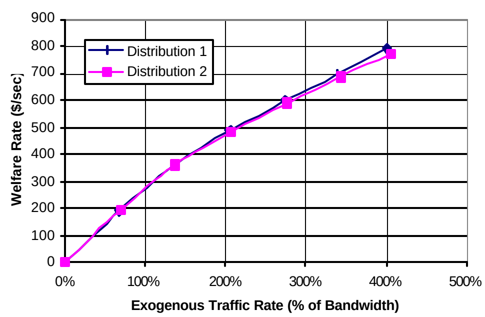
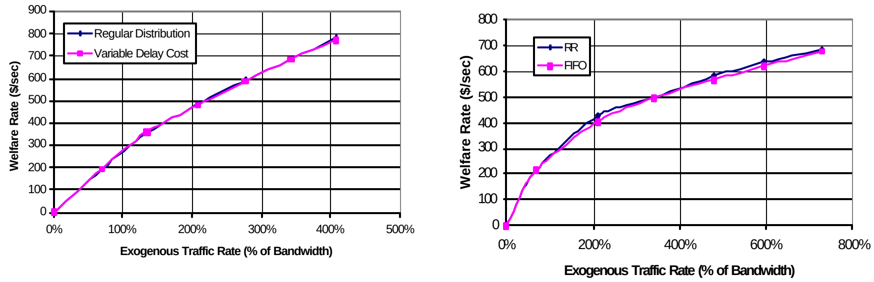


Figure 6: Welfare rates differentiated in standard deviations of job value and delay cost

The experiment demonstrated that there is no significant difference when using a variable job value or a delay cost coefficient correlating to job size. Welfare rate curves from two differentiated schemes in delay cost coefficient, one with a fixed distribution and the other with a variable mean correlating to job size, match well (Figure 7a). The same outcome is obtained from the scheme using variable job value correlating to job size. The curvature of welfare rate curves from the schemes using variable job value is more concave compared to previous welfare rate curves (Figure 7b).



(a) Welfare rates from two schemes differentiated in the delay cost correlating to the job size.

(b) Welfare rates from FIFO and non-prioritized RR scheduling schemes using variable job value correlating to the job size.

Figure 7: Welfare rates from the schemes using variable parameters correlating to job size

In summary we can conclude that traffic pricing can effectively improve VPN bandwidth service benefits for a wider range of schemes using different job values and delay cost distributions.

4.3 Human Subject Based Experiments

The direct experiments using human subjects allows us to test the performance of traffic pricing system closer to real environment and also verify the parameters used in agent-based experiments. The experiments are performed in two steps: first, subjects are used to access the experiment system using both pricing and non-pricing schemes; then, the data flow patterns recorded during the process are replayed in agent-based experiment mode. All experiments in this phase use non-prioritized RR bandwidth scheduling. Figure 8 shows the outcomes from different setups of experiments. Curves *Agent_Pricing*, and *Agent_Non_P* are the welfare rates - total service benefits per second - from experiments using the agent-based traffic generation. The curves indicate that dynamic pricing improves bandwidth service welfare. This is the same result as the one from previous experiments. Similarly, welfare rates, with regard to different levels of traffic arrival rates, obtained from direct experiments using pricing (curve *Human_Pricing*) are relatively higher than the welfare rates from direct experiments without pricing (curve *Human_Non_P*). This strongly suggests the consistency between direct experiments and agent-based experiments.

It is noticeable that direct experiments show relatively lower welfare rates than the welfare rates from the same schemes as conducted in agent-based experiment. Nevertheless, curve *Human_Pricing* is almost parallel to *Agent_Pricing*, and *Human_Non_P* is also almost parallel to *Agent_Non_P*, indicating the compatibility of the outcomes from two types of experiments. Two possible reasons may cause the welfare rate observed from direct experiments lower than that from agent-based experiments: 1) the human errors and occasional irrationality, and 2) the different exogenous flow of potential traffic between two types of experiments. Roth (1996) commented "that human behavior deviates in systematic ways from the idealized behavior attributed to expected utility maximizers in particular, and to 'rational economic man' in general". A direct experiment using human subjects is closer to the real world, but may introduce errors caused by subjects' misbehaviors.

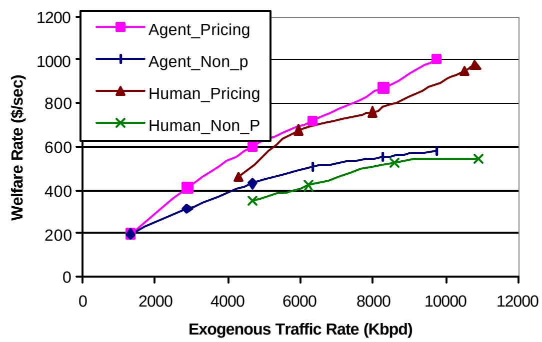


Figure 8. Welfare Rates (Bandwidth: 1544 Kbps)

To investigate the issue, a group of simulations are carried out using the same exogenous traffic generation process as the direct experiments. The difference is in that the experimental system with the replaying setup makes optimum submission decisions to maximize the expected user utility for every request. The service welfare rates from these replay experiments are as good as those from experiments using agent-generated network traffic, and are better than those from direct experiments producing these exogenous traffic patterns (Table 1). The last column of the table shows that the welfare rate ratio between a direct experiment and the agent-based experiment replaying its traffic pattern, both using traffic pricing scheme, is about 86%. The same ratio for the experiments without using pricing is higher, being 93.3% in a sample case. Thus, we can conclude that this loss is due to user mistakes rather than exogenous traffic differences.

Table 1: Welfare Rate Comparison

Schemes	Exog. Traffic Rate (kbps)	Welfare Rate ($/sec) (Direct)	Welfare Rate ($/sec) (Replay)	Welfare Rate Ratio (Direct / Replay)
Dataset 1 (Pricing)	10601	948	1098	86.4%
Dataset 2 (Pricing)	7750	763	893	85.4%
Dataset 3 (Non_pricing)	4695	361	387	93.3%

In checking the size distribution of data flows that are incurred by submitted requests it is impressive that human subjects sometimes tend to make "regrettable" request submission decisions: submitting a request having a positive expected net value but realizing a negative outcome, or dropping a request because of its negative expected net value which would be actually positive if the request were serviced. Each individual subject's performance, in sense of the surplus ratio in two types of experiments, varies from as high as 91.4% to as low as 71.3%, indicating the existence of a subjective factor affecting subjects' performance. In another aspect, as a consequence of the imprecise information available to human subjects and more precise information for computer agents to make decisions, human subjects have a higher average surplus forecasting error with a much higher standard deviation of the error rate, where error rate is the ratio between the error of the estimated user surplus and the average user surplus.

5. SUMMARY

This chapter studies the VPN traffic-pricing problem targeting at the practical application with three foci: a transaction-level pricing architecture with a traffic proxy server-based implementation scheme, a dynamic traffic-pricing model for the VPN, and VPN traffic-pricing experiments using a prototype system called VTPES. The transaction-level pricing is proposed for the implementation of the VPN pricing system, taking advantage of VPN's user-account management features. This job-oriented basis is exploited to derive the dynamic VPN traffic-pricing model. The theoretical work is based on the GSW model with two important extensions for the VPN—it is tailored to fit packet-switched networks, typically the Internet, and it uses round-robin bandwidth scheduling in addition to FIFO bandwidth scheduling. The pipeline effect of packet-switched networks results in the dominance of the bottleneck Internet connection in a VPN route over total throughput time as well as the bandwidth price, which can be controlled by the TPS.

In a simplified case, the total price a job pays for bandwidth services can be approximated by a single price at the bottleneck gateway to reduce the complexity of implementation. We revealed that RR scheduling possesses useful implementation features such as allowing a consistent unit price for different types of jobs. The experimental outcomes from both direct experiments and simulations strongly support the theoretical result. The three aspects of the research in VPN traffic pricing jointly provide a complementary set of solutions for the VPN traffic control problem.

There are four apparent limits of our model and implementation scheme. The first one is that a practical gateway may use more than one scheduling algorithm for its bandwidth allocation tasks, for example, a mixed FIFO and

RR scheduling. Deriving an analytical form of the price formula for a real gateway is very difficult. Even though the TPS-based solution proposed in this paper may bypass this problem, the mutual effect between the TPS and the VPN gateway in data flow control remains untouched.

The second limit is that the proposed VPN pricing model is designed for the problem of elastic traffic where the optimal objective function counts the total throughput time as the only QoS feature. In pricing real-time traffic, the important factors for evaluating service quality include jitter, i.e., the variation of available bandwidth; committed minimum bandwidth; and maximum allowable bandwidth. Quantifying these features is difficult and more dimensions will hence need to be introduced into the model.

The third limit is inherent in the assumption that the job arrival is a Poisson process. Paxson and Floyd (1995) revealed that the packet arrival pattern in the Internet is not a Poisson process. It is apparent that even if the interval of any two consecutive jobs is exponentially distributed the arrival of packets is not necessarily a Poisson process because the number of packets in a job is a random variable and these packets come in batches. Nonetheless, Alok, Stahl, and Whinston (1999) reported that the pricing formula based on Poisson arrivals works well even when arrivals are fractal.

The fourth limit is also very critical: we calculate the service delay as a linear function of traffic load at a network node without considering the non-linear effect when congestion happens. Once a network node's buffer overflows, it tosses some packets. This will stimulate reactions of the flow control mechanism to resolve the problem. In this case, the delay of an affected data flow will no longer be linear with its size. Since our experiments have shown that a VPN with the pricing mechanism performs better than a non-pricing VPN when the infinite queue is assumed, the former must also be superior to the non-pricing system with limited queue capacity. In the next research phase, we may let the model cope with the congestion in two aspects.

The above limits reflect the gap between a theoretical research and the practical application. As long as we have a suitable traffic-pricing architecture, the issue raised from the practical system can be solved accordingly with proper patches.

Acknowledgement

This research has been sponsored by Intel Corp., Lucent Technologies Inc., and Mannesman Ag.

[1] One of recent focuses in this research direction is the application of network traffic pricing to *differentiated services* (Blake et al, 1998), for example, *dynamic capacity contracting* by Kalyanaroman, S., T. Ravichandran and R. Norsworthy (1998). This indicates the effort in reducing the gap between theory and practice in this area.

[2] A public network, in contrast, does not provide user account management functions, so traffic pricing would be practically very difficult.

[3] A typical technical explanation of the cause of congestion is queue overflow that results in packet tosses and hence incurs retransmissions. The consequence is that the situation gets worse and transmissions are overwhelmingly delayed.

[4] *Job* is considered a legacy term from the old batch-processing system. We adopt this name to keep the consistency with other papers, having used it previously.

[5] The definition of *transaction* by whatis.com (http://www.whatis.com): "In computer programming, a transaction usually means a sequence of information exchange and related work (such as database updating) that is treated as a unit for the purposes of satisfying a request and for ensuring database integrity. For a transaction to be completed and database changes to made permanent, a transaction has to be completed in its entirety."

[6] The extension is straightforward under the assumption of sequential processing of the multiple data flows, since the costs are additive. The proof is somewhat tedious and trivial, so it is not added as another appendix but is available upon request.

[7] The latter is normally defined as *latency*, which ranges between 30-200 ms depending on the transmission distance and the quality of the circuit. As a typical example, on July 6, 2001, the latency of AT&T WorldNet in a 1925-mile of end-to-end connection is 32 ms in a 24-hour average (http://ratings.matrix.net/).

[8] Each channel has its own optimum rental price. Subscript s, which indexes the channel, is negligible by default, because the bottleneck price of a VPN Internet channel is only affected by the local traffic. If a data flow is not indexed by a channel label s, it should be interpreted as the traffic through the pricing channel being discussed. Therefore, we discard the subscript s from all notations.

References:

Allman, M., V. Paxson, and W. Stevens. "TCP Congestion Control," Internet RFC2581 (1999), http://www.ietf.cnri.reston.va.us/rfc/rfc2581.txt.

Blake, S. et al, "An Architecture for Differentiated Services," Internet RFC2475 (1998), http://www.ietf.cnri.reston.va.us/rfc/rfc2475.txt.

Braden B. et al. "Recommendations on Queue Management and Congestion Avoidance in the Internet," Internet RFC2309 (1998), http://www.ietf.cnri.reston.va.us/rfc/rfc2309.txt.

Clark, D. D. "Adding Service Discrimination to the Internet," *Telecommunication Policy* 20 (1996) 169-181.

Demers, A., S. Keshav, and S. Shenker. "Analysis and Simulation of a Fair Queuing Algorithm," *Internetworking: Research and Experience* 1 (1990) 3-26.

Floyd, S. "TCP and Explicit Congestion Notification," *ACM Computer Communications Review* 24 (1994) 10-23.

Floyd, S., and Kevin Fall, "Promoting the Use of End-to-End Congestion Control in the Internet," *IEEE/ACM Transactions on Networking* 7:4 (1999) 458-472.

Floyd, S., and V. Jacobson. "Random Early Detection (RED) Gateways for Congestion Avoidance," *IEEE/ACM Transactions on Networking* 1:4 (1993) 397-413.

Gupta, A. et al. "Streaming the Digital Economy: How to Avert a Tragedy of the Commons," *IEEE Internet Computing* 1 (1997) 38-46.

Gupta, A., D. O. Stahl, and A. B. Whinston. "A Stochastic Equilibrium Model of Internet Pricing," *Journal of Economic Dynamics and Control* 21 (1997) 697-722.

Gupta, A., D. O. Stahl, and A. B. Whinston. "The Economics of Network Management," *Communications of the ACM* 42 (1999) 57-63.

Jacobson, V., and M. J. Karels. "Congestion Avoidance and Control," *SIGCOMM'88*, (1988).

Kalyanaroman, S., T. Ravichandran and R. Norsworthy. "Dynamic Capacity Contracting: A Framework for Pricing the Differentiated Services Internet," (1998), http://www.ecse.rpi.edu/Homepages/shivkuma/research/papers/icepaper.html.

Kleinrock, L. *Queueing Systems,* Vol. I & II, John Wiley & Sons, 1975.

Kosiur, D. *Building and Managing Virtual Private Netowrks*, John Wiley & Sons, 1998.

Lazar, A. A., and Nemo Semret. "Design, Analysis and Simulation of the Progressive Second Price Auction for Network Bandwidth," *8th International Symposium on Dynamic Games and Applications* (1998).

Li, Mingzhi et al. "Estimating Internet Users' Demand Characteristics," forthcoming in *Journal of Computational Economics* (2000), http://crec.bus.utexas.edu/works/articles/demand.pdf.

Lin, Z. et al. "Exploring Traffic Pricing for the Virtual Private Network," *The Proceedings of WITS'99*, (1999).

Lin, Z. et al. "Exploring Traffic Pricing for the Virtual Private Network," forthcoming in *Information Technology and Management*, (2001).

Lin, Z., D. O. Stahl, and A. B. Whinston. "A Traffic-pricing Model for the Packet-switching Network with Prioritized Round-robin Queueing," *INFORMS-CIST'00*, (2000).

MacKie-Mason, J., and H. Varian. "Pricing the Internet in Public Access to the Internet," In B. Kahin and J. Keller (eds.), Englewood Cliffs, NJ:Prentice-Hall, 1995.

Nagle, J. "Congestion Control in TCP/IP Internetworks," Internet RFC 896 (1984), http://www.ietf.cnri.reston.va.us/rfc /rfc896.txt.

Odlyzko, A. M. "A Modest Proposal for Preventing Internet Congestion," *DIMACS Technical Report 97-68* (1997), http://www.dimacs.rutgers.edu/TechnicalReports/.

Oppliger, R. *Internet and Intranet Security,* Artech House, 1998.

Paxson, V., and S. Floyd. "Wide-Area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking* 3(3) (1995).

Rigney, C. et al. "Remote Authentication Dial In User Service (RADIUS)," Internet-draft (2000), http://www.ietf.org/internet-drafts/draft-ietf-radius-radius-v2-06.txt.

Rigney, C., and Livingston. "RADIUS Accounting," Internet RFC 2139 (1997), http://www.ietf.org/rfc/rfc2139.txt.

Roth, A. E. “Comments on Tversky’s ‘Rational Theory and Constructive Choice’,” *The Rational Foundations of Economic Behavior*, K. Arrow, E. Colombatto, M Perlman, and C. Schmidt (eds.), Macmillan, (1996) 198-202.

Shenker, S. et al. “Pricing in Computer Networks: Reshaping the Research Agenda,” *Telecommunication Policy* 20 (1996) 183-201.

Stahl, D. O., and A. B. Whinston. “A General Economic Equilibrium Model of Distributed Computing,” in W. W. Cooper and A. B. Whinston (eds.), *New Decisions in Computational Economics* (1994) 175-189.

Sterman, J. “Testing Behavioral Simulation Models by Direct Experiment,” *Management Science* 33(12), 1987, 1572-1592.

Stevens, W. “TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms,” Internet RFC2001 (1997), http://www.ietf.cnri.reston.va.us/rfc/rfc2001.txt.

L. Tesfatsion, “Agent-Based Computational Economics: A Brief Guide to the Literature,” January, 2000. http://www.econ.iastate.edu/tesfatsi/aceintro.pdf

Authors’ Bios:

Zhangxi Lin is currently an assistant professor of information system in the College of Business Administration at Texas Tech University. He obtained a M.Eng. degree in computer science from Tsinghua University in 1982, a M.S degree in economics in 1996 and a Ph.D. degree in information systems in 1999 from the University of Texas at Austin. His research interests include network traffic management and congestion-based pricing, economic modeling of online auctions, economic experimental system design and experimental testing, and the Internet and its application in China. He is a guest professor/researcher for four universities in China.

Peng Si Ow is currently a consultant on the deployment of innovations from R&D into operations. She began her work at IBM where she was involved in developing and deploying state of the art production planning and control systems into the field. Her work later extended from manufacturing systems to the deployment of corporate intranets and more recently to medical imaging systems. She has a Ph.D. in Systems Sciences from Carnegie Mellon University.

Dale Stahl is currently the Malcolm Forsman Centennial Professor of Economics in the Department of Economics at the University of Texas, Austin. He obtained B.S. and M.S. degrees in electrical engineering from M.I.T. in 1969 and a Ph.D. from the University of California at Berkeley in 1981. His early research was in general equilibrium theory and price adjustment mechanisms, followed by research in strategic pricing and advertising. After many years of contributions to game theory, he began investigations and experimental testing of theories of bounded rationality and learning. He has also become involved in economic analyses of the Internet infrastructure with emphasis on congestion-based pricing. His publications include a book on electronic commerce, and 60 articles in top journals.