Performance Study of Congestion Price based Adaptive Service

Xin Wang, Henning Schulzrinne
Dept. of Computer Science
Columbia University
1214 Amsterdam Avenue
New York, NY 10027
xwang@ctr.columbia.edu, schulzrinne@cs.columbia.edu

Abstract— **In a network with enhancements for QoS support, pricing of network services based on the level of service, usage, and congestion provides a natural and equitable incentive for applications to adapt their sending rates according to network conditions. In this paper, we first propose a dynamic, congestion-sensitive pricing algorithm, and also develop the demand behavior of adaptive users based on a physically reasonable user utility function. We then develop a simulation framework to compare the performance of a network supporting congestion-sensitive pricing and adaptive reservation to that of a network with a static pricing policy. We also study the stability of the dynamic pricing and reservation mechanisms, and the impact of various network control parameters. The results show that the congestion-sensitive pricing system takes advantage of application adaptivity to achieve significant gains in network availability, revenue, and user-perceived benefit relative to the fixed-price policy. Congestion-based pricing is stable and effective in limiting utilization to a targeted level. Users with different demand elasticity are seen to share bandwidth fairly, with each user having a bandwidth share proportional to its relative willingness to pay for bandwidth. The results also show that even a small proportion of adaptive users may result in a significant performance benefit and better service for the entire user population - both adaptive and non-adaptive users. The performance improvement given by the congestion-based adaptive policy further improves as the network scales and more connections share the resources.**

I. Introduction

Resource reservation and adaptive services are two basic models for allocating resources to multimedia applications. Compared to resource reservation, the adaptation approach has the advantage of better utilizing available network resources, which change with time. But if network resources are shared by competing users, users of rate-adaptive applications do not have any incentive to scale back their sending rate below their access bandwidth, since selfish users will generally obtain better quality than those that reduce their rate. In a network with enhancements for QoS support, pricing of network services based on the level of service, usage, and congestion provides a natural and equitable incentive for applications to adapt their sending rates according to network conditions. Increasing the price during congestion gives the application an incentive to back-off its sending rate and at the same time allows an application with more stringent bandwidth and QoS requirements to maintain a high quality by paying more.

Earlier we presented a Resource Negotiation and Pricing (RNAP) protocol and architecture [20]. RNAP enables the user to select from available network services with different QoS properties and re-negotiate contracted services, and enables the network to dynamically formulate service prices and communicate current prices to the user. Our framework offers a middle ground, where resources are reserved, but resource commitments are made only for short intervals, instead of indefinitely. Prices may vary for each interval, encouraging applications to adjust their resource demands to network congestion. Unlike best-effort adaptive approaches, applications are guaranteed resources and there is no assumption that applications are cooperative. Our model allows the network operator to create different trade-offs between blocking admissions and raising congestion prices to prevent overload.

In this paper, we first propose a dynamic, congestion-sensitive pricing algorithm, and also develop the demand behavior of adaptive users based on a physically reasonable user utility function. We then develop a simulation framework to compare the performance of a network supporting congestion-sensitive pricing and adaptive reservation to that of a network with a static pricing policy. We also study the stability of the dynamic pricing and reservation mechanisms. We try to answer questions such as how much do the network and users gain in terms of revenue and perceived benefit (or value-for-money) under the dynamic and static systems, and how do various pricing and adaptation parameters affect the functioning of the dynamic system. The simulation framework is based on the RNAP model, but we try to derive results and conclusions applicable to static and congestion-driven, dynamic pricing schemes in general.

In Section II of this paper, we present a brief outline of the RNAP framework. In Section III, we discuss various network pricing models and their suitability. We discuss in detail a volume-based, congestion-sensitive pricing strategy, also presented earlier in [20]. In Section IV, we consider user adaptation in response to congestion-dependent pricing. We present a physically reasonable form of user utility function, and derive a specific demand function for a given network price based on this utility function. In section V, we describe the simulation topology and parameters, and performance metrics. In Section VI, we discuss simulation experiments in detail, and in Section VII, we describe some related work. We summarize our findings in Section VIII.

II. Resource Negotiation through RNAP

In the RNAP framework [20][22], we assume that the network makes services with certain QoS characteristics available to user applications, and charges prices for these services that, in general, vary with the availability of network resources. Network resources are obtained by user applications through negotiation between the Host Resource Negotiator (HRN) on the user side, and a Network Resource Negotiator (NRN) acting on behalf of the network. The HRN negotiates on behalf of one or multiple applications belonging to a multimedia system. In an RNAP session, the NRN periodically provides the HRN updated prices for a set of services. Based on this information and current ap-

plication requirements, the HRN determines the current optimal transmission bandwidth and service parameters for each application. It re-negotiates the contracted services by sending a *Reserve* message to the NRN, and receiving a *Commit* message as confirmation or denial.

The HRN only interacts with the local NRN. If its application flows traverse multiple domains, resource negotiations are extended from end to end by passing RNAP messages hop-by-hop from the first-hop NRN until the destination network NRN, and vice versa. End-to-end prices and charges are computed by accumulating local prices and charges as *Quotation* and *Commit* messages travel hop-by-hop upstream towards the HRN.

III. Pricing Strategies

A few pricing schemes are widely used in the Internet today [16]: access-rate-dependent charge (AC), volume dependent charge (V), or the combination of the both (AC-V). An AC charging scheme is usually one of two types: allowing unlimited use, or allowing limited duration of connection, and charging a per-hour fee for additional connection time. Similarly, AC-V charging schemes normally allow some amount of volume to be transmitted for a fixed access fee, and then impose a per-volume charge. Although time-of-day dependent charging is commonly used in telephone networks, it is not used in the current Internet.

User experiments [3] indicate that usage-based pricing is a fair way to charge people and allocate network resources. Both connection time and the transmitted volume reflect the usage of the network. Charging based on connect-time only works when resource demands per time unit are roughly uniform. Since this is not the case for Internet applications and across the range of access speeds, we only consider volume-based charging.

In this paper, we study two kinds of volume-based pricing: a fixed-price (FP) policy with a fixed unit volume price, and a congestion-price-based adaptive service (CPA) in which the unit volume price has a congestion-sensitive component. We now describe the latter system in more detail, and also present a generic pricing framework to accommodate the different pricing models.

A. Fixed Pricing

In the fixed price model, the network charges the user per volume of data transmitted, independent of the congestion state of the network. The per-byte charge can be the same for all service class ("flat", FP-FL), depend on the service class (FP-PR), depend on the time of day (FP-T) or a combination of time-of-day and service class (FP-PR-T). Since our focus is on the congestion-based dynamic pricing, and the fixed-price system serves as a reference, we assume a general fixed pricing structure that represents all the four categories depending on the underlying network service infrastructure and the service provider's business model.

B. Congestion-based Pricing

If the price does not depend on the congestion conditions in the network, customers with less bandwidth-sensitive applications have no motivation to reduce their traffic as network congestion increases. As a result, either the service request blocking rate will increase sharply at the call admission control level, or the packet dropping rate will increase greatly at the queue management level. Having a congestion-dependent component in the service price provides a monetary incentive for adaptive applications to adapt their service class and/or sending rates according to network conditions. In periods of resource scarcity, quality sensitive applications can maintain their resource levels by paying more, and relatively quality-insensitive applications will reduce their sending rates or change to a lower class of service.

The total price of CPA will be composed of a component that depends on congestion and a fixed volume-based charge. Thus, with four variations on the fixed volume-based charge outlined above, we have the pricing models CP-FL, CP-PR, CP-T, CP-PR-T. This is summarized in Table 1.

We assume that routers support multiple service classes and that each router is partitioned to provide a separate link bandwidth and buffer space for each service, at each port. We consider one of the classes. We use the framework of the competitive market model [19]. The competitive market model defines two kinds of agents: consumers and producers. Consumers seek resources from producers, and producers create or own the resources. The exchange rate of a resource is called its price. The routers are considered the producers and own the link bandwidth and buffer space for each output port. The flows (individual flows or aggregate of flows) are considered consumers who consume resources. The congestion-dependent component of the service price is computed periodically, with a price computation interval τ. The total demand for link bandwidth is based on the aggregate bandwidth reserved on the link for a price computation interval, and the total demand for the buffer space at an output port is the average buffer occupancy during the interval. The supply bandwidth and buffer space need not be equal to the installed capacity; instead, they are the targeted bandwidth and buffer space utilization. The congestion price will be levied once demands exceeds a provider-set fraction of the available bandwidth or buffer space. We now discuss the formulation of the fixed charge, which we decompose into *holding charge* and *usage charge*, and the formulation of the *congestion charge*.

B.1 Usage Charge

The usage charge is determined by the actual resources consumed, the average user demand, the level of service guaranteed to the user, and the elasticity of the traffic. For example, on a per-byte basis, best-effort traffic will cost less than reserved, non-preemptable CBR traffic. The usage price (p_u) will be set such that it allows a retail network to recover the cost of the purchase from the wholesale market, and various static costs associated with the service. The usage_charge $c_u(n)$ for a period n in which $V(n)$ bytes were transmitted is given by:

$$c_u(n) = p_u V(n) \tag{1}$$

B.2 Holding Charge

The holding charge can be justified as follows. If a particular flow or flow-aggregate does not utilize the resources (buffer space or bandwidth) set aside for it, we assume that the scheduler allows the resources to be used by excess traffic from a lower level of service. The holding charge reflects revenue lost by the provider because instead of selling the allotted resources at the usage charge of the given service level (if all of the reserved resources were consumed) it sells the reserved resources at the usage charge of a lower service level. The holding price (p_h) of a service class is therefore set to be proportional to the difference between the usage price for that class and the usage price for the next lower service class. The holding price can be represented as:

$$p_h^i = \alpha^i(p_u^i - p_u^{i-1}), \quad (2)$$

where α^i is a scaling factor related to service class i. The holding_charge $c_h(n)$ when the customer reserves a bandwidth $R(n)$ is given by:

$$c_h(n) = p_h R(n)\tau \quad (3)$$

where τ is the duration of the period. $R(n)$ can be a bandwidth requirement specified explicitly by the customer, or estimated from the traffic specification and service request of the customer.

B.3 Congestion Charge

The congestion price for a service class is calculated as an iterative tâtonnement process [19]:

$$p_c(n) \quad = \quad \min[\{p_c(n-1) + \sigma(D,S)(D-S)/S, 0\}^+, p_{max}] \quad (4)$$

where D and S represent the current total demand and supply respectively, and σ is a factor used to adjust the convergence rate. σ may be a function of D and S; in that case, it would be higher when congestion is severe. The router begins to apply the congestion charge only when the total demand exceeds the supply. Even after the congestion is removed, a non-zero, but gradually decreasing congestion charge is applied until it falls to zero to protect against further congestion. In our simulations, we also used a price adjustment threshold parameter θ to limit the frequency with which the price is updated. The congestion price is updated if the the calculated price increment exceeds $\theta p_c(n-1)$.

The maximum congestion price is bounded by the p_{max}. When a service class needs admission control, all new arrivals are rejected when the price reaches p_{max}. If p_c reaches p_{max} frequently, it indicates that more resources are needed for the corresponding service.

For a period n, the total congestion charge is given by

$$c_c(n) = p_c(n)V(n). \quad (5)$$

Based on the price formulation strategy described above, a router arrives at a cost structure for a particular RNAP flow or flow-aggregate at the end of each price update interval. The total charge for a session is given by

$$c_s = \sum_{n=1}^{N} [p_h R(n)\tau + (p_u + p_c(n))V(n)] \quad (6)$$

where N is the total number of intervals spanned by a session.

In some cases, the network may set the usage charge to zero, imposing a holding charge for reserving resources only, and/or a congestion charge during resource contention. Also, the holding charge would be set to zero for services without explicit resource reservation, for example, best effort service.

C. A Generic Pricing Structure

We have now discussed several approaches to charging the customer for network services, and described one of them (usage sensitive congestion based pricing) in detail. The following generic equation represents the charge incurred by a customer for a single billing cycle in all these cases:

$$\begin{aligned} cost \quad = \quad & c_{ac}(R_{ac}) + p(R_{ac})(t - T_m)^+ + \sum_{i=1}^{I} \sum_{n=1}^{N_b} [p_h^i(n) R^i(n)\tau \\ & + (p_u^i(n) + p_c^i(n))V^i(n)](V^i - V_m^i)^+ \end{aligned} \quad (7)$$

Here I is the number of service classes in the network, i represents a particular service class, c_{ac} represents the access rate dependent fixed charge, $p(R_{ac})$ is the unit time connection price charged for the excess time above a contracted free of charge duration T_m, t is the total duration of a billing cycle, N_b is the number of price update intervals during a billing cycle, V^i is the total volume of class i traffic transmitted during the billing cycle, V_m^i is the volume of traffic from class i that is free of charge, and other parameters have the same meaning as in Section III-B. Multiple service classes may be used during a billing cycle, either at different times, or simultaneously for different co-existing applications (for example, belonging to a teleconference application). Generally, p_h and p_u usually vary only slowly, on the order of hours, while p_c will change much more rapidly. For the different charging modes discussed in previous sections, equation 7 contain different items shown in table I.

As equation 7 shows, a volume based charging scheme can also have an access charge component. In that case, the network may either specify a certain threshold volume below which only the access charge applies, or alternatively, specify a threshold rate R_m (less than or equal to the access link rate), so that the volume threshold for a single price updation period is of the form $R_m \times \tau$. Setting a contracted threshold rate instead of a threshold volume encourages users to smooth out their traffic, and thus allows resources to be provisioned more economically.

In our simulations, we implement both a congestion-dependent pricing model for the CPA service, and a fixed price model for the FP service. Since we do not consider service class interactions, and do not consider time-of-day dependence, in effect, we implement the CP-FL and FP-FL models. However, we believe the results from the CPA and FP to be applicable to all the CP and FP pricing models, as well as the access charge inclusive CP model, in a lot of important respects, since the most important and influential feature of the models is the presence or absence of congestion-dependent pricing.

IV. User Adaptation

In a network with congestion dependent pricing and dynamic resource negotiation (through RNAP or some other signaling protocol), *adaptive* applications with a budget constraint will adjust their service requests in response to price variations. In this section, we discuss how a set of user applications performing a given task (for example, a video conference) adapt their sending rate and quality of service requests to the network in response to changes in service prices, so as to maximize the benefit or *utility* to the user, subject to the constraint of the user's budget.

Although we focus on adaptive applications as the ones best suited to a dynamic pricing environment, the RNAP framework does not require adaptation capability. Applications may choose services that provide a fixed price and fixed service parameters during the duration of service. Generally, the long-term average cost for a fixed-price service will be higher, since it uses network resources less efficiently. Alternatively, applications may use a service with usage-sensitive pricing, and maintain a high QoS level, paying a higher charge during congestion.

We consider a set of user applications, required to perform a task or *mission*. The user would like to determine a set of transmission parameters (sending rate and QoS parameters) from which it can derive the maximum benefit, subject to his budget. We assume that the user defines quantitatively, through a *utility function*, the perceived monetary value (say, 15 cents/minute) provided by the that set of transmission parameters towards com-

Charging Scheme	Access	Connection Time	Holding	Usage	Congestion	Class-Based	Time-dependent
AC	yes	yes					
FP-FL	optional		yes	yes			
FP-PR	optional		yes	yes		yes	
FP-T	optional		yes	yes			yes
FP-PP-T	optional		yes	yes		yes	yes
CP-FL	optional		yes	yes	yes		
CP-PR	optional		yes	yes	yes	yes	
CP-T	optional		yes	yes	yes		yes
CP-PR-T	optional		yes	yes	yes	yes	yes

TABLE I

THE CHARGING STRUCTURE OF DIFFERENT SCHEMES

pleting the mission.

Consumers in the real world generally try to obtain the best possible "value" for the money they pay, subject to their budget and minimum quality requirements; in other words, consumers may prefer lower quality at a lower price if they perceive this as meeting their requirements and offering better value. Intuitively, this seems to be a reasonable model in a network with QoS support, where the user pays for the level of QoS he receives. In our case, the "value for money" obtained by the user corresponds to the surplus between the utility $U(\cdot)$ with a particular set of transmission parameters (since this is the perceived value), and the cost of obtaining that service. The goal of the adaptation is to maximize this surplus, subject to the budget and the minimum and maximum QoS requirements.

We now consider the simultaneous adaptation of transmission parameters of a set of n applications performing a single task. The transmission bandwidth and QoS parameters for each application are selected and adapted so as to maximize the mission-wide "value" perceived by the user, as represented by the surplus of the *total utility*, $\hat{U}$, over the total cost C. We can think of the adaptation process as the allocation and dynamic re-allocation of a finite amount of resources between the applications.

In this paper, we make the simplifying assumption that for each application, a utility function can be defined as a function only of the transmission parameters of that application, independent of the transmission parameters of other applications. Since we consider utility to be equivalent to a certain monetary value, we can write the total utility as the sum of individual application utilities :

$$\hat{U} = \sum_i [U^i(x^i)] \tag{8}$$

where x^i is the transmission parameter tuple for the i_{th} application. The optimization of surplus can be written as

$$\begin{aligned} max & \sum_i [U^i(x^i) - C^i(x^i)] \\ & \text{s. t.} \sum_i C^i(x^i) \leq b \\ & x^i_{min} \leq x^i \leq x^i_{max} \end{aligned} \tag{9}$$

where x^i_{min} and x^i_{max} represent the minimum and maximum transmission requirements for stream i, and C^i is the cost of the type of service selected for stream i at requested transmission parameter x^i.

In practice, the application utility is likely to be measured by user experiments and known at discrete bandwidths, at one or a few levels of loss and delay, possibly corresponding to a subset of the available services; at the current stage of research, some possible services are guaranteed [18] and controlled-load service [23] under the int-serv model, Expedited Forwarding (EF) [10] and Assured Forwarding (AF) [9] under diff-serv. In this case, it is convenient to represent the utility as a piecewise linear function of bandwidth (or a set of such functions). A simplified algorithm is proposed in [21] to search for the optimal service requests in such a framework.

We can make some general assumptions about the utility function as a function of the bandwidth, at a fixed value of loss and delay. A user application generally has a minimum requirement for the transmission bandwidth. It also associates a certain minimum value with a task, which may be regarded as an "opportunity" value, and this is the perceived utility when the application receives just the minimum required bandwidth. The user terminates the application if its minimum bandwidth requirement can not be fulfilled, or when the price charged is higher than the opportunity value derived from keeping the connection alive. Also, user experiments reported in the literature [13][2] suggest that utility functions typically follow a model of diminishing returns to scale, that is, the marginal utility as a function of bandwidth diminishes with increasing bandwidth. Hence, a utility function can be represented in a general form as:

$$U(x) = \max(U_0 + w \log \frac{x}{x_m}, 0) \tag{10}$$

where x_m represents the minimum bandwidth the application requires, w represents the sensitivity of the utility to bandwidth, and U_0 is the monetary "opportunity" that the user perceives in the application.

When the utilities of all the applications are represented in the format of equation 10, the optimization process for a system with multiple applications can be represented as:

$$\begin{aligned} \max & \sum_j [U_0^j + w^j \log \frac{x^j}{x_m^j} - p^j x^j] \\ & \text{s. t.} \sum_j p^j x^j \leq b \\ & \text{and} \quad x^j \geq x_m^j, \forall j \end{aligned} \tag{11}$$

If the user can obtain the optimal bandwidth for the system at a cost below his budget, then the user demand that maximizes the perceived surplus can be shown to be:

$$x^j = \frac{w^j}{p^j} \tag{12}$$

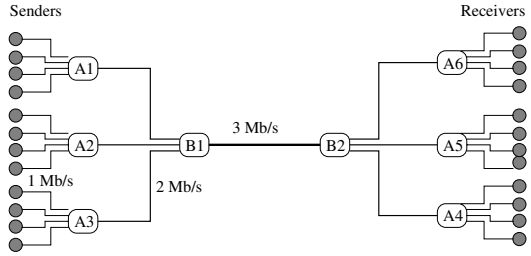

Fig. 1. Simulation network topology 1

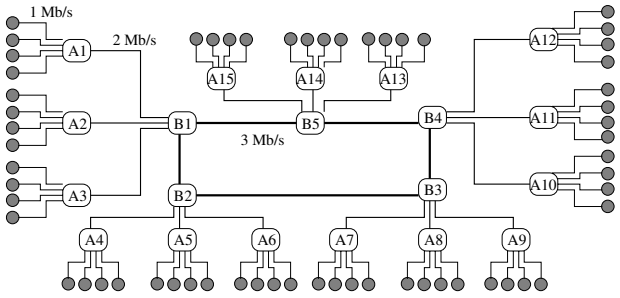

Fig. 2. Simulation network topology 2

Hence, w^j represents the money a user would spend based on its perceived value for an application.

If the total bandwidth a system can obtain is bounded by the budget, then the optimal demand becomes:

$$x^j = \frac{b\frac{w^j}{\sum_l w^l}}{p^j} \tag{13}$$

Therefore, when the budget is a constraint, each application in a system receives a share based on the user's perceived value of this application.

V. Simulation Model

In this section, we describe our simulation model for the CPA and FP policies. The policies are simulated at the call level, that is, we consider user resource contention due to the total user requested bandwidth exceeding the provisioned system bandwidth, rather than due to the burstiness of user traffic. Depending on the service type and network infrastructure, the network may learn user resource requirements explicitly through a signaling protocol, or implicitly by traffic measurement. We simulate explicit resource reservation and price signaling through RNAP.

We used the *network simulator* [1] environment to simulate two different network topologies, shown in Fig. 1 and Fig. 2. Topology 1 contains two backbone nodes, six access nodes, and twenty-four end nodes. Topology two contains five backbone nodes, fifteen access nodes, and sixty end nodes. Topology two was also used in [6]. All links are full duplex and point-to-point. The links connecting the backbone nodes are 3 Mb/s, the links connecting the access nodes to the backbone nodes are 2 Mb/s, and the links connecting the end nodes to the access nodes are 1 Mb/s. At each end node, there is a fixed number N_s of sending users. We use topology 1 in most of our simulations to allow us to simulate congestion from a single bottleneck node, and only use topology 2 to illustrate the CPA performance under a more general network topology in Section VI-G.

User requests are generated according to a Poisson arrival process and the lifetime of each flow is exponentially distributed with an average length of 10 minutes. In topology 1, users from the sender side independently initialize unidirectional flows towards randomly selected receiver side end nodes. At most $12N_s$ flows (48 sessions with N_s set to 4) can run simultaneously in the whole network. In topology 2, all the users initialize unidirectional flows towards randomly selected end nodes. At most $60N_s$ users (360 sessions with N_s set to 6) are allowed to run simultaneously in the whole network.

The users are assumed to have the general form of the utility function shown in Section IV. w, the elasticity factor, (and also the user's willingness to pay) is uniformly distributed between \$0.125/min and \$0.375/min for a 64kb/s bandwidth. The opportunity cost U_0 is set to the amount a user is willing to pay for its minimum bandwidth requirement, and is hence given by $U_0 = p_{high} \times x_{min}$, where p_{high} is the maximum price the user will pay before his connection is dropped. Users re-negotiate their resource requirements with a period of 30 seconds in all the experiments.

The unit bandwidth price charged by the FP policy, and the unit bandwidth usage price charged by CPA, p_u, are both set to \$0.15/min for 64 kb/s transmission. The holding price p_h in the CPA policy is assumed to be zero, since all simulations are currently performed within a single service class, and interactions between service classes are not considered. The targeted link utilization of the CPA policy is 90% unless otherwise specified, and congestion pricing is applied when instantaneous usage exceeds this threshold. The price adjustment procedure is also controlled by a pair of parameters, the price adjustment step σ from equation 4 and the price adjustment threshold parameter θ, defined in Section III-B.3. Unless otherwise specified, values of $\sigma = 0.06$ and $\theta = 0.05$ are used.

In the simulation, we show the performance of the system for a range of *offered loads*. The offered load is defined as the ratio between the total user resource requirement at the bottleneck, and the bottleneck capacity. Under the FP policy, the total user resource requirement is also the actual resource demand from all the users. Under the CPA policy, the total user resource requirement is what the total resource demand would be if there were no resource contention at the bottleneck and the network did not impose an additional congestion-dependent price.

Both economic and engineering performance metrics are of interest in our study. We define the following engineering performance metrics:

- *Bottleneck bandwidth utilization:* The average bandwidth utilization at the bottleneck node is measured by averaging the reserved bandwidth (expressed as a ratio of the link capacity) over all negotiation periods.
- *User request blocking probability:* The user request blocking probability is the percentage of user reservation requests being denied by the system, due to insufficient provisioned resources. Unsuccessful re-negotiation during an ongoing session is not considered as a block, and the old resource reservation will be maintained upon failure of re-negotiation.

We also define the following economic performance metrics:

- *Average and total user benefit:* The user benefit is the perceived value a user obtains through a transmission of a certain bandwidth (which may vary during the transmission due to adaptation by the user) and of a certain duration, calculated using the user's utility function. Clearly, the user

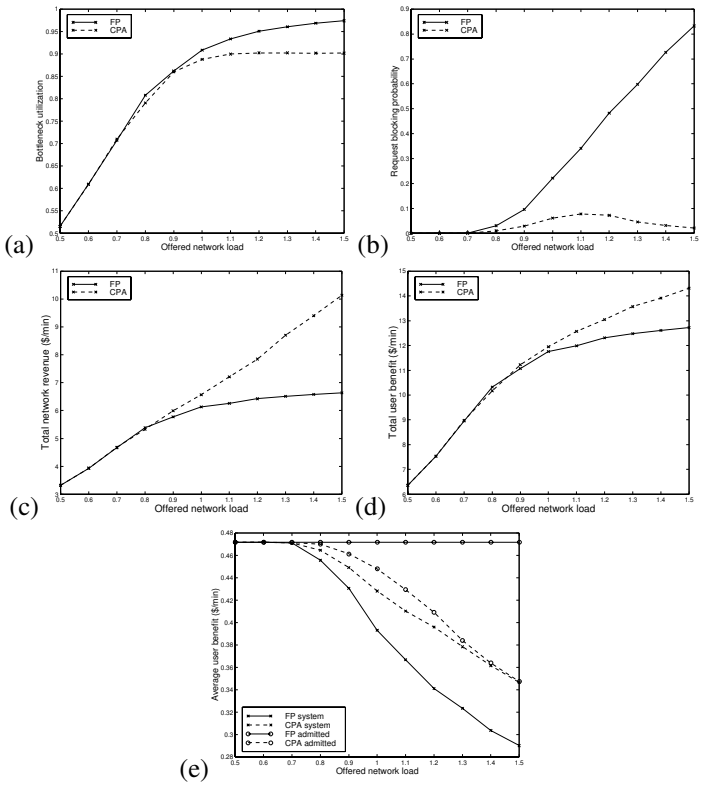

Fig. 3. Performance metrics of CPA and FP policies as a function of offered load: (a) bottleneck utilization; (b) blocking probability; (c) total network revenue; (d) total user benefit; (e) average user benefit.

obtains no benefit if its connection request is blocked. The average user benefit is the average of perceived benefits obtained by all the users, and the total user benefit is the sum of perceived benefits obtained by all the users.

Price: We monitor the end-to-end price quoted by the network during a simulation as a measure of the stability of the price adjustment / user adaptation process.

User Charge: A user is charged based on its bandwidth requirements during a user session and the corresponding price quoted by the network.

Network revenue: Network revenue is the total charge paid to the network for all the admitted requests during a simulation.

VI. Results and Discussion

In this section, we show simulation results from the set of experiments described in section Section V.

A. FP Policy versus CPA Policy

We first compare the performance under the FP policy and the CPA policy, with the default conditions specified in Section V. Figs. 3 (a)-(d) depict the results of the simulations:

- Fig. 3 (a) shows the variation of the utilization as a function of the offered load, expressed as a fraction of the link capacity. The network utilization under FP policy increases continuously with the increase of offered load. The utilization of CPA policy initially increases with the increase of the offered as expected, and then saturates at the targeted reservation level of 0.9 as the offered load increases beyond a threshold 1.1. This is as expected, since the objective of the CPA policy is to provide the users the incentive to back off their individual resource requirements in period of resource contention so that the total resource demand remain within the targeted level.
- Both policies admit all connections until the total link capacity is saturated. Fig. 3 (b) indicates that the blocking probability of FP scheme increases almost linearly as the offered load increases beyond 0.9, while the blocking rate of CPA increases initially and then starts to decrease after reaching a maximum at offered load 1.1. This is because the price adjustment step is proportional to the excess bandwidth above the targeted utilization and increases progressively faster with offered load at higher loads, and the user bandwidth request decreases proportionally with the price according to the general utility function of Section IV. The blocking probability of FP policy is almost 40 times larger than that of the CPA policy at the heaviest load.
- Fig. 3 (c) compares the network revenue under both FP and CPA policies as a function of the offered load. The FP policy flattens out after the onset of request-blocking, indicating that the average number of accepted connections increases slowly beyond this point. With the CPA policy, the revenue increases more than linearly after the network utilization saturates at the targeted level. The loss of revenue due to the scaling down of individual bandwidth requests is more than offset by gains due to the admission of more connections and the increase in the congestion price.
- Fig. 3 (d) shows that the user benefit flattens out for both policies after the onset of request blocking. The total benefit gained under CPA is higher than that under FP beyond this point, and the difference increases as the offered load increases. As illustrated in Section IV, there is a potential opportunity cost associated with a request being blocked. The decrease in perceived benefit per connection of CPA due to the reduction of bandwidth is offset by the increase in the number of admitted connections, each of which receives an "opportunity". In effect, the CPA policy allows the network bandwidth to be used more efficiently under high loads.
- Fig. 3 (e) shows the average perceived benefit per user against offered load. For the FP policy, individual user requests do not depend on the offered load, and consequently, the average benefit per *admitted* user is independent of offered load. However, a progressively smaller fraction of users is admitted by the FP policy as offered load increases. Therfore, the average perceived benefit across all users decreases sharply with the load. The CPA has a much smaller blocking probability, which gives a higher average perceived benefit as load increases. This should serve as an incentive for users to choose the CPA policy over the FP policy.

We now consider the dynamics of the system price, user bandwidth demand, and user expenditure during the simulation. The results are shown in Figs. 4 (a)-(e).

- Figs. 4 (a) and (b) show the dynamic variation of the system price and user bandwidth demand respectively at three different levels of offered load. The bandwidth demand is shown for an "average" user, that is, one whose minimum and maximum bandwidth requirements are averages of the corresponding requirements of the user population. The price and bandwidth are nearly static at a load of 0.8, and are adjusted more frequently at higher offered loads, due to

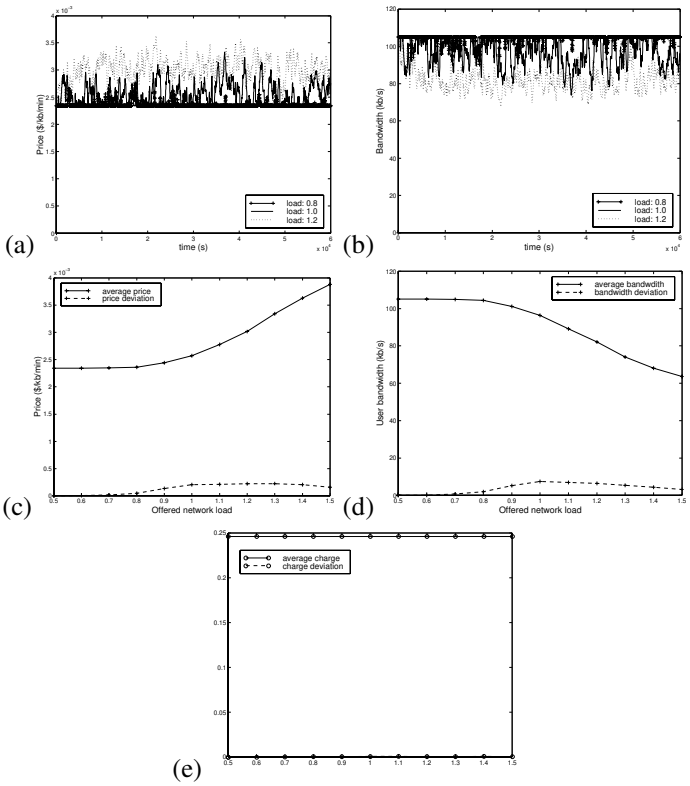

Fig. 4. System dynamics under CPA: variation over time of system price (a), and average user demand (b), at on offered load of 1.2; time-average and standard deviation of system price (c), average user demand (d), and average user expenditure (e), plotted against offered load.

the more frequent arrival and departure of users.

- Figs. 4 (c) and (d) show the average and standard deviations of the system price and user bandwidth demand as a function of the offered load. The standard deviation in both figures shows the same trend as the blocking speed of Fig. 3 (b), an increase to a certain level and then a decrease. Initially, the price and demand deviations increase as load increases due to the more aggressive congestion control. At heavy loads, the increased multiplexing of user demand smooths the total demand, and therefore reduces fluctuations in the price.
- From the perspective of the user, the session cost (expenditure) and application level QoS performance are the most significant metrics. Fig. 4 (e) shows when the users adapt under the example utility function of Section IV, the user can operate at a stable expenditure, and therfore under a fixed budget, meeting one of the fundamental goals of demand adaptation.

The total variation in price over a range of loads also depends on the basic usage price and holding price values, which should be set to reflect the long term user demand for different service classes, so that demand fluctuations above the congestion threshold are short-term and infrequent, and congestion pricing is only occasionally employed to smooth out traffic peaks. We are still studying the interaction of long term network resource provisioning with the short term network resource negotiation.

The results in this section indicate that the CPA policy takes advantage of application adaptivity for significant gains in network availability, revenue, and perceived user benefit, relative to the fixed-price policy. The congestion-based pricing is stable

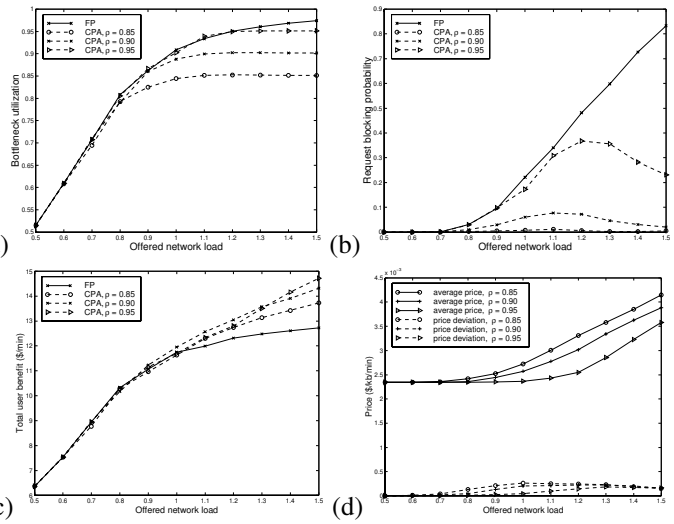

Fig. 5. Performance of CPA and FP policies at different values of target congestion control threshold ρ: (a) bottleneck utilization; (b) blocking probability; (c) total user benefit; (d) time-average and standard deviation of system price under CPA.

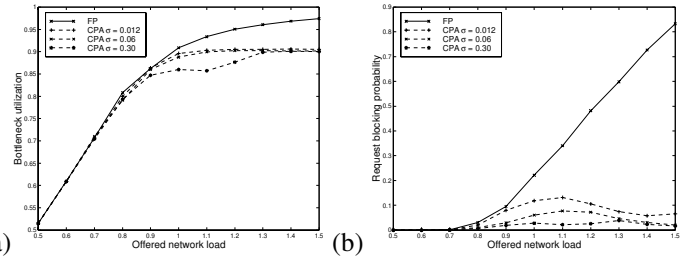

Fig. 6. Performance of CPA and FP at different values of σ: (a) bottleneck utilization; (b) blocking probability.

and effective. If the nominal (un-congested) price is set to correctly reflect long-term user demand, the congestion-based pricing should effectively limit short-term fluctuations in load.

B. Variations of Network Control Parameters

In this section, we study the impact of certain network control parameters on the network and user metrics. The parameters are: the congestion control threshold (or targeted link utilization) ρ beyond which the congestion-dependent price component is imposed; the price scaling factor σ, used to control the rate at which a congested link is brought back to the targeted utilization; and the price adjustment threshold θ, which limits the frequency with which the price is updated. The parameters are varied one at a time.

In Fig. 5, the user benefit decreases if the target utilization is set either too low or too high. Also, with too low a target, demand fluctuations are higher, while too high a targeted level results in a high blocking rate. Increasing the price scaling factor σ (which affects the speed of reaction to congestion) significantly reduces the blocking probability (Fig. 6). However, too large a value of σ results in network under-utilization at offered loads close to the target utilization, and also results in large network dynamics. If the price adjustment threshold parameter θ is set too high, there is no meaningful price adjustment and adaptive action. Below a certain level, further reductions in θ do not give performance benefits or disadvantages (Fig. 7).

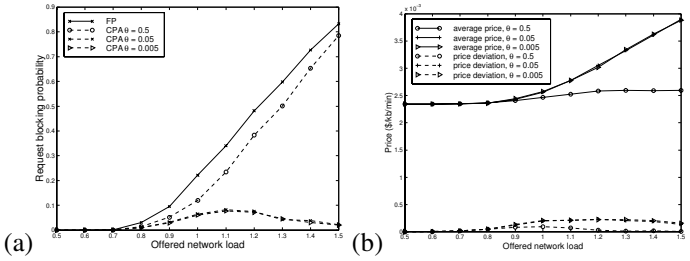

Fig. 7. Performance of CPA and FP at different values of θ: (a) blocking probability; (b) time-average and standard deviation of system price under CPA.

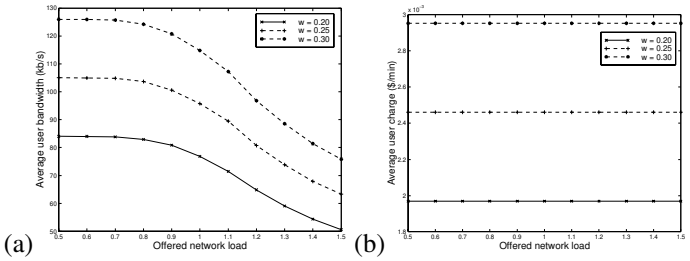

Fig. 8. Effect of the elasticity factor w on bandwidth allocation and user expenditure: (a) average bandwidth reserved by users with the three different values of w; (b) average expenditure of users with the three different values of w

C. Effect of User Demand Elasticity

In this experiment, we study the effect of the user demand elasticity factor w on the system performance. A smaller value of w corresponds to a more elastic demand, since the bandwidth-dependent component of the utility is smaller, and the user can reduce its bandwidth request in response to a price increase with only a small decrease in utility. As explained in Section IV, w also represents a user's willingness to pay for bandwidth.

Users with different demand elasticity are seen to share bandwidth fairly, with each user having a bandwidth share proportional to its relative willingness to pay for bandwidth (Fig. 8). In effect, users with more stringent bandwidth requirements choose to pay a higher charge and "borrow" bandwidth from users with more elastic requirements when the network is congested.

D. Effect of Session Multiplexing

We vary the number of customers sharing a system and evaluate the effect of the increased multiplexing of session requests under both CPA policy and FP policy as the number of sessions is increased. We keep the network topology and user utility distributions unchanged, but scale the link capacity proportionally with the maximum number of flows.

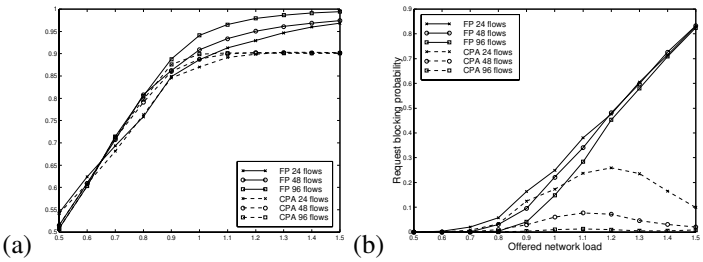

Fig. 9. Performance of CPA and FP with different number of customers sharing the system: (a) bottleneck utilization; (b) blocking probability.

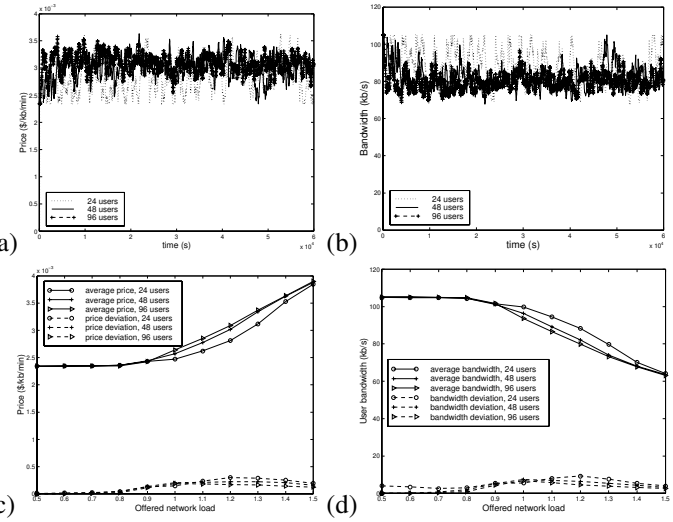

Fig. 10. System dynamics with different number of customers sharing the same bottleneck: variation over time of system price (a), and average user demand (b), at an offered load of 1.2; time-average and standard deviation of system price (c) and average user demand (d), plotted against offered load.

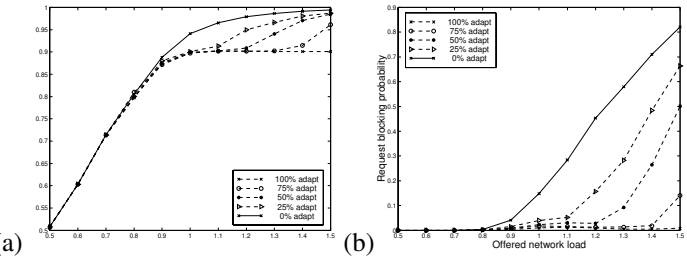

Fig. 11. Performance of CPA when only some of the users adapt their bandwidth requests: (a) bottleneck utilization; (b) blocking probability.

Fig. 9 (a) shows that the overall link utilization under FP increases as the number of connections increases, at a given offered load. The link utilization under CPA also increases with the number of flows at moderate to high loads, but the utilization is eventually limited to the targeted level. Fig. 9 (b) shows that, as the number of connections increases, the blocking probability decreases under both FP policy and CPA policies. This is because that the larger number of connections lead to better traffic multiplexing and hence more efficient use of network bandwidth. However, the improvement is much more pronounced under the CPA policy than under the FP policy, particularly when the network is saturated. Under CPA, the blocking rate with 96 connections is up to 50 times smaller than that with 24 connections.

Fig. 10 depicts the price and demand dynamics as the network scales. Figs. 10 (a) and (b) show that the frequency of price and demand adjustment do not change appreciably with the number of connections. As expected, both price and user bandwidth demand become smoother as more users share the network, and this is confirmed by the smaller standard deviations shown in Figs. 10 (c) and (d).

The results in this section indicate that performance of the CPA policy further improves as the network scales and more connections share the resources.

E. Adaptive and Non-adaptive Users

In this section, we consider the environment where some users adapt their bandwidth requests under the CPA policy, while oth-

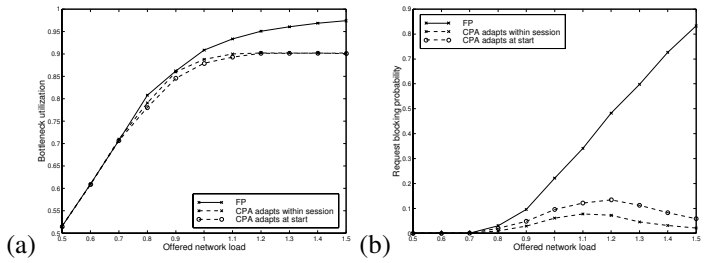

Fig. 12. Performance when CPA users select bandwidth only at session set-up, compared with performance when they continue to adapt during the session (a) bottleneck utilization; (b) blocking probability.

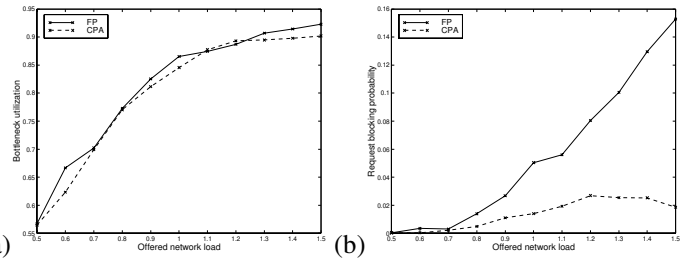

Fig. 13. Performance metrics of CPA and FP policies as a function of offered load using topology 2: (a) bottleneck utilization; (b) blocking probability.

ers maintain fixed service requests even when the congestion price is imposed. The latter group represents users with a willingness to pay that is high enough to maintain their maximum bandwidth requirements even at the highest price charged by the network. In this set of simulations, we restrict the maximum price so that the price does not increase without bound when all of the users are non-adaptive.

The results show that even a small proportion of adaptive users may result in a significant performance benefit and better service for the entire user population - both adaptive and non-adaptive users - particularly up to a certain threshold load. The total user-perceived benefit is seen to increase with the proportion of adaptive users (Fig. 11).

We should also expect CPA to have an additional inherent advantage over the FP policy even when most of the users are non-adaptive. In reality, the usage price shown in Section III-B would reflect the estimated long-term network load. The congestion price would be only used to smooth out temporary peaks, and the general usage pattern would result in optimal utilization at the offered usage price. However, a vendor charging a static price (FP) would need to charge a certain premium above this optimal price, as a risk premium, while the CPA policy allows the vendor to operate around the optimal price and use congestion pricing to protect against demand peaks.

F. Session Adaptation and Adaptive Reservation

Under RNAP, applications can either pick a bandwidth when starting a session and keep that bandwidth during the session or adjust its resource demands during each negotiation interval. We refer to these modes as initial adaptation and ongoing adaptation, respectively.

Fig. 12 (a) shows that initial adaptation results in a slightly lower network utilization at moderate-to-high loads, about 3-5% smaller than the utilization under ongoing adaptation. This is because if a session arrives during a traffic peak, it will request a smaller bandwidth, which will not be scaled back after the the demand is driven down. Fig. 12 (b) shows that as expected, adaptation during a session allows for more efficient bandwidth usage and the blocking probability is reduced by half.

G. CPA Performance with Traffic Interactions from Different Paths

In the experiments above, we studied the performance of CPA when the traffic shares a common bottleneck. In this section, we assume network topology 2 in Fig. 2, with the potential for multiple bottlenecks to exist, and for these bottlenecks to interact.

In the simulation, traffic is generated symmetrically from all users, as described in Section 5. The five backbone links are the potential bottleneck links. Note that in reality, the backbone links are normally over-provisioned. We target the backbone links to be bottlenecks only for the convenience of simulation. We monitor the utilization at one of the backbone links, and calculate all the other parameters across the whole network. Fig. 13 (a) and (b) shows that both the utilization and blocking probability show trends similar to those for a single bottleneck, except that the variation of the utilization and blocking probability is not as smooth due to the coupling of the traffic between different paths.

H. Other Mechanisms to Reduce Network Variations

The user adaptation behavior also influences the variation in bandwidth seen by application as well as the overall network behavior. A user can, for example, only requests a change in bandwidth if the price change exceeds a given range. This reduces both the frequency of bandwidth adjustment and the user surplus. The initial adaptation described in Section VI-F is the limit case where user reservation reflects only the price quoted at the beginning of the session.

A somewhat similar scenario can be envisioned in a core network, in which bandwidth reservation is carried out by other network providers rather than by individual users. In this case, the providers can change their bandwidth requests in multiples of a large block of bandwidth, only when the user flow-level demand to the customer providers changes by a certain increment. This can reduce both network dynamics and signaling overhead in the core network, and has been discussed in greater detail in [20].

VII. Related Work

Microeconomic principles have been applied to various network traffic management problems. The studies in [15][14][11][7] are based on a maximization process to determine the optimal resource allocation such that the utility (a function that maps a resource amount to a satisfaction level) of a group of users is maximized. These approaches normally rely on a centralized optimization process, which does not scale. Also, some of the algorithms assume some knowledge of the user's utility curves by the network and truthful revelation by users of their utility curves, which may not be practical.

In [5][4][7][8][17], the resources are priced to reflect demand and supply. The pricing model in these approaches is usage-sensitive - it has been shown that usage-sensitive pricing results in higher utilization than traditional flat (single) pricing [5]. Some of these methods are limited by their reliance on a well-defined statistical model of source traffic, and are generally not intended to adapt to changing traffic demands.

In general, the work cited above differs from ours in that it does not enter into detail about the negotiation process and the network architecture, and mechanisms for collecting and communicating locally computed prices. Some of the work also as-

sumes immediate adjustment of the price in response to the network dynamics, or require the user to maintain a static demand until a optimal price is found, which is not practical. Our work is concerned with developing a flexible and general framework for resource negotiation and pricing and billing, and evaluating the performance benefits of congestion-sensitive pricing and adaptation through simulations, decoupled from specific network service protocols. Our work can therefore be regarded as complementary to some of the cited work.

In [12], a charging and payment scheme for RSVP-based QoS reservations is described. A significant difference from our work is the absence of an explicit price quotation mechanism - instead, the user accepts or rejects the estimated charge for a reservation request. Also, the scheme is coupled to a particular service environment (int-serv), whereas our goal is to develop a more flexible negotiation protocol usable with different service models.

VIII. Conclusion

We have considered a framework for incentive-driven rate and QoS adaptation. In the framework, users respond actively to changes in price signaled by the network by dynamically adjusting network resource usage by the application, so as to maximize the perceived utility relative to the price, subject to user budget and QoS constraints. We have discussed different pricing models, and outlined a dynamic, congestion-sensitive pricing algorithm. We have also described the user demand behavior based on a physically reasonable user utility characteristic.

The main focus of this paper has been the simulation of the above framework. Through simulations, we have compared the performance of a network under the congestion price based adaptation policy (CPA) with that under a fixed price based policy (FP). We have also studied the stability of the adaptation process, and nature of network dynamics, under the CPA policy. In general, CPA policy takes advantage of application adaptivity for significant gains in network availability, revenue, and perceived user benefit (in terms of the user utility functions), relative to the fixed-price policy. The congestion-based pricing is stable and effective in limiting utilization to a targeted level. If the nominal (un-congested) price is set to correctly reflect long-term user demand, the congestion-based pricing should effectively limit short-term fluctuations in load.

We have investigated the impact of various network control parameters on the network and user metrics. The user benefit decreases if the target utilization is set either too low or too high. Also, with too low a target, demand fluctuations are higher, while too high a targeted level results in a high blocking rate. Increasing the price scaling factor σ (which affects the speed of reaction to congestion) significantly reduces the blocking probability. However, too large a value of σ results in network underutilization at offered loads close to the target utilization, and also results in large network dynamics. If the price adjustment threshold parameter θ is set too high, there is no meaningful price adjustment and adaptive action. Below a certain level, further reductions in θ do not give performance benefits or disadvantages.

Users with different demand elasticity are seen to share bandwidth fairly, with each user having a bandwidth share proportional to its relative willingness to pay for bandwidth. The results also show that even a small proportion of adaptive users may result in a significant performance benefit and better service for the entire user population - both adaptive and non-adaptive users. The performance improvement given by the CPA policy further improves as the network scales and more connections share the resources.

In this paper, we assume that users do not have the option of choosing a different path or provider, reflecting current network reality. However, pricing in the presence of competition or alternative paths remains an interesting open issue.

References

[1] Network simulator - ns (version 2) . http://www-mash.CS.Berkeley.EDU/ns/.

[2] Watson A. and M. A. Sasse. Evaluating audio and video quality in low-cost multimedia conferencing systems. In *Interacting with Computers*, volume 8, pages 255–275, 1996.

[3] J. Altmann, B. Rupp, and P. Varaiya. Internet user reactions to usage-based pricing. In *Proceedings of the 2nd Berlin Internet Economics Workshop (IEW '99)*, Berlin. Germany, May 1999.

[4] N. Anerousis and A. A. Lazar. A framework for pricing virtual circuit and virtual path services in atm networks. In *ITC-15*, December 1997.

[5] R. Cocchi, S. Shenker, D. Estrin, and L. Zhang. Pricing in computer networks: Motivation, formulation, and example. In *IEEE/ACM Transactions on Networking*, December 1993.

[6] M. Creis. Rsvp/ns: An implementation of rsvp for the network simulator ns-2. Document, Computer Science Department, University of Bonn.

[7] D. F. Ferguson, C. Nikolaou, and Y. Yemini. An economy for flow control in computer networks. In *Conference on Computer Communications (IEEE Infocom), (Ottawa, Canada)*, April 1989.

[8] E. W. Fulp and D. S. Reeves. Distributed network flow control based on dynamic competive markets. In *Proceedings International Conference on Network Protocol (ICNP'98)*, October 1998.

[9] J. Heinanen. Assured forwarding PHB group. Internet Draft, Internet Engineering Task Force, August 1998. Work in progress.

[10] V. Jacobson, K. Nichols, and K. Poduri. An expedited forwarding PHB. Internet Draft, Internet Engineering Task Force, February 1999. Work in progress.

[11] H. Jiang and S. Jordan. A pricing model for high speed networks with guaranteed quality of service. In *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, March 1996.

[12] M. Karsten, J. Schmitt, L. Wolf, and R. Steinmetz. An embedded charging approach for rsvp. 1998.

[13] C. Lambrecht and O. Verscheure. Perceptual quality measure using a spatio-temporal model of human visual system. In *Proc. of IS&T/SPIE*, February 1996.

[14] C. Lee, J. Lehoczky, R. Rajkumar, and D. Siewiorek. A quality of service negotiation procedure for distributed multimedia presentational applications. In *Proceedings of the Fifth IEEE International Symposium On High Performance Distributed Computing (HPDC-5)*, 1996.

[15] J. F. MacKie-Mason and H. Varian. Pricing congestible network resources. September 1995.

[16] P. Reichl, S. Leinen, and B. Stiller. A practical review of pricing and cost recovery for internet services. In *Proc. of the 2nd Internet Economics Workshop Berlin (IEW '99)*, Berlin, Germany, May 1999.

[17] J. Sairamesh. Economic paradigms for information systems and networks. In *PhD thesis*, October 1997.

[18] S. Shenker, C. Partridge, and R. Guerin. Specification of guaranteed quality of service. Request for Comments (Proposed Standard) 2212, Internet Engineering Task Force, September 1997.

[19] Hal Varian. *Microeconomic Analysis*. W.W. Norton & Co, 1993.

[20] X. Wang and H. Schulzrinne. RNAP: A resource negotiation and pricing protocol. In *International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'99)*, pages 77–93, Basking Ridge, New Jersey, June 1999.

[21] X. Wang and H. Schulzrinne. Adaptive reservation: A new framework for multimedia adaptation. In *IEEE International Conference on Multimedia and Expo (ICME'2000)*, New York, NY, USA, July 2000.

[22] X. Wang and H. Schulzrinne. An integrated resource negotiation, pricing, and qos adaptation framework for multimedia applications. In *IEEE Journal on Selected Areas in Communications*, volume 18, 2000.

[23] J. Wroclawski. Specification of the controlled-load network element service. Request for Comments (Proposed Standard) 2211, Internet Engineering Task Force, September 1997.