Adaptive Reservation: A New Framework for Multimedia Adaptation

X. Wang, H. Schulzrinne
Columbia University
xwang@ctr.columbia.edu, schulzrinne@cs.columbia.edu

Abstract—

Research on multimedia adaptation usually assumes undifferentiated, best-effort service from the network, and relies on signaling mechanisms such as packet loss rates for feedback. These methods result in unfairness to adaptive applications in the presence of aggressive, non-adaptive applications. A network with enhancements for QoS support, and usage and QoS dependent pricing, can use pricing as a natural incentive to drive adaptive behavior by applications. In this work, we present a framework for the adaptation of multimedia application sending rate and/or choice of network services in response to dynamic network pricing and changes in application requirements. The adaptation is based on the (monetary) value of a service as perceived by the user, relative to the price charged by the network. Experimental results show that perceived-value-based adaptation allows bandwidth to be shared fairly among competing users. When network resources are scarce, bandwidth is shown to be distributed among competing applications (and among media streams belonging to a single multimedia system) according to their relative elasticity of demand, indicated by the sensitivity of the perceived value to the bandwidth.

I. Introduction

The development and use of distributed multimedia applications are growing rapidly. These applications usually require a minimum Quality of Service (QoS) from the network, in terms of throughput, packet loss, delay, and jitter. To address these problems, one approach is to enhance the network with mechanisms such as resource reservation, admission control, and special scheduling mechanisms. Another approach is to adjust the bandwidth used by an application according to the existing network conditions [1], relying on signaling mechanisms such as packet loss rates for feedback. 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 [2]. 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.

In earlier work, we presented a Resource Negotiation and Pricing (RNAP) protocol and architecture [2]. 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. In this paper, we propose some approaches towards adaptation of (multimedia) application sending rate and/or choice of network services in response to the incentive provided by dynamic network pricing. We discuss how to maximize user satisfaction in such an environment, subject to the constraints imposed by the minimum and maximum QoS requirements of the application, and the available budget. We also discuss the allocation of resources to component streams (audio, video, etc.) belonging to a multimedia system, for across-the-system maximization of value to the user. We present experimental results demonstrating important features of the adaptation process.

In section II of this paper, we briefly describe the RNAP architecture, as an example of the environment in which incentive-driven adaptation takes place. In section III, we discuss incentive-driven adaptation algorithms, and in section IV, we present some experimental results.

II. Resource Negotiation through RNAP

In this section, we briefly describe the RNAP protocol and architecture [2], as a typical framework within which incentive-driven adaptation by the user takes place.

In the RNAP framework, we assume that the network makes services with certain QoS characteristics available to the 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 application 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.

III. User Adaptation

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, under the constraint of the user's budget.

A. The Utility Function

We consider a set of user applications, required to perform a task or *mission*, for example, audio, video, and whiteboard applications for a video-conference. The *Reserve* request from the user specifies certain transmission parameters for each application. In general, the transmission parameters are the sending rate, as well as QoS parameters, usually loss and delay. The user must define quantitatively, through a *utility function*, the value provided by the corresponding network resource allocation towards completing the mission. The utility function is therefore a function in a multi-dimensional space, with each dimen-

This work was supported in part by Hughes Research lab and by Ericsson.

sion representing a single transmission parameter allocation for a particular application.

A.1 Utility as Perceived Value

Clearly, the utility of a transmission depends on its quality as perceived by the user. However, since the user is paying for the transmission, it appears reasonable to define the utility as the *perceived value* of that quality to the user. An audio transmission requiring a certain sending rate and certain bounds on the end-to-end delay and loss rate may be worth 10 cents/minute to the user. The perceptual value is strongly correlated to the perceptual quality, but is not exactly the same. A pair of audio transmissions encoded identically and with the same transmission QoS parameters also have the same perceived quality, but their perceived values may differ according to the application requirements.

The measurement of subjective quality of multimedia transmissions has been reported by a number of researchers. Generally, these experiments were intended to derive the Mean Opinion Score (MOS), which is measured as an average perceptive quality across a number of test subjects. However, in our framework, perceived value very strongly reflects individual user preferences, and the application task being performed. We therefore consider it likely that an user application will have one or more of the following features:

- allow user to customize utility function(s);
- allow user to define "scenario"-specific utility functions; a particular scenario may be selected by the user during a session, or may be deduced by the application based on user actions;
- allow user to specify a certain time-dependence of utility function.

A.2 Utility as a Function of Bandwidth

It is likely that only a few alternative services will be available to a multimedia application on the Internet - at the current stage of research, some possible services are guaranteed [3] and controlled-load service [4] under the int-serv model, Expedited Forwarding (EF) [5] and Assured Forwarding (AF) [6] under diff-serv. A particular user application would be able to choose from a small subset of the available services. Each such service would probably provide some qualitative or quantitative guarantee for loss and delay. It seems likely, therefore, that the user would develop an utility function as a function of the transmission bandwidth (which in turn would depend on specific encoding parameters such as frame rate, quantization, etc.), at different discrete levels of loss rate and delay.

We can make some general assumptions about the utility function as a function of the bandwidth, at a fixed value of loss and delay. The application has a minimum transmission bandwidth, and the utility is zero for bandwidth below this threshold. Also, user experiments reported in the literature 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 and eventually goes to zero, defining a maximum QoS requirement.

B. Application Adaptation

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.

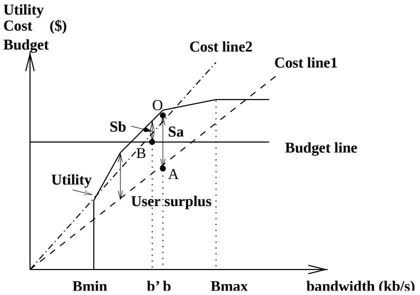

Fig. 1. A perceived value based rate adaptation model

We first consider the adaptation strategy of a single application when its utility is a function only of bandwidth (at a fixed loss and delay). We then discuss the adaptation strategy when the utility is function of multiple transmission parameters (bandwidth, loss and delay). Finally, we consider the problem of maximizing the *mission-wide* utility of a system comprising multiple applications performing a certain task. We assume the applications belong to a single user.

B.1 Adaptation of Single Application with Fixed Transmission Quality

If the quality of transmission is fixed (a particular delay and loss), the application utility (that is, the user-perceived value) increases monotonically with the bandwidth. Hence the maximization problem for the user can be written as:

$$\begin{aligned} \max\ & [U(x) - C(x)] \\ \text{s. t.}\ & C(x) \leq b \\ & x_{min} \leq x \leq x_{max}, \end{aligned} \tag{1}$$

where x is the bandwidth under consideration, $C(x)$ is the cost for the requested bandwidth, b is the budget of user, x_{min} is the minimum bandwidth requirement, and x_{max} represents the maximum bandwidth requirement. Note that U, b and c are in units of money/time.

One way of carrying out this optimization is to fit the utility function to a closed form function. The optimal solution is then obtained by using Kuhn-Tucker conditions for a maximum subject to inequality constraints. As mentioned earlier, the application utility is likely to be measured by user experiments and known at discrete bandwidths. In this case, it is convenient to represent the utility as a piecewise linear function, as shown in Fig. 1. The figure also assumes a constant unit bandwidth cost C, so that the cost-vs-bandwidth is a straight line with slope equal to C. The budget is shown as a horizontal line passing intercepting the cost/utility axis. From the figure, it is evident that the optimal bandwidth is **either** the segment end-point with the highest surplus, if this end-point meets the budget constraint (b in Fig. 1 case A), **or else** the bandwidth corresponding to the intersection point of the cost line with the budget line (b' in Fig. 1 case B).

B.2 Adaptation of Single Application over Multiple Transmission Parameters

We now consider the maximization of the application surplus over a set of transmission parameters (usually, the bandwidth, loss rate and delay). The objective function is as shown earlier in equation 1, but x, x_{min} and x_{max} are now vectors corresponding to the set of transmission parameters. If a complete quality of service parameter space is considered, the searching cost can be prohibitive. As discussed earlier, however, we believe it is likely that the application utility will take the form of a small set of utility versus transmission bandwidth functions, each at a different level of loss rate and delay, corresponding to a particular service. In this case, the optimization routine is as follows:

1. For each available service, use the corresponding utility versus bandwidth function to determine the optimal bandwidth, as in Section III-B.1.
2. Select the service which gives the highest surplus at its optimal bandwidth.

B.3 Simultaneous Adaptation of Multiple Applications Performing Single Task

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.

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 :

$$\hat{U} = \sum_i [U^i(x^i)], \tag{2}$$

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

$$\begin{aligned} max \sum_i [U^i(x^i) - C^i(x^i)] \\ \text{s. t.} \sum_i C^i(x^i) \leq b \\ x^i_{min} \leq x^i \leq x^i_{max}, \end{aligned} \tag{3}$$

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.

As earlier, we can decompose a single utility function $U^i(x^i)$ into a set of service-specific utility functions which are functions only of bandwidth, each corresponding to a particular delay and loss provided by a particular service. Clearly, several combinations of services (and hence, service-specific utility functions) are possible. We first consider one particular combination of service-specific utility functions. Let the utility of an application i be defined at L^i bandwidth levels. The utility at each level is u^i_l ($l = 1, 2, ..L^i$), and the utility function is piecewise linear. Segment l (the straight line between levels l and $l+1$) has a slope k^i_l. The optimal transmission parameter set for a particular combination of service-specific utility functions is then determined as follows:

1. From the utility function for each application i, determine the segment end-point $l_{opt}(l = 1, 2, ..L^i)$, with bandwidth B^i_{opt}, at which the surplus (utility minus cost) is maximized for that application. Let the cost of the targeted bandwidth be $C^i_{opt}(B^i_{opt})$.
2. If the total expenditure needed for the system $\sum_i C^i_{opt}(B^i_{opt})$ exceeds the total system budget, go to step 3, else stop.
3. From all the applications that receive service at level $l_{opt} > l_{min}$, find the application i_{victim} with the smallest slope in the surplus ($u^i_l - C^i_l$) from level l_{opt} to $l_{opt} - 1$ (this corresponds to the smallest sensitivity of application surplus to a reduction in bandwidth). Reduce the current bandwidth allocation for this application to the next lower bandwidth level ($l_{opt} = l_{opt} - 1$).
4. If the total system expenditure remains greater than the system budget, go back to step 3. If there is excess budget, allocate the excess budget to the current victim application (from step 3) to acquire as much bandwidth as permitted by the budget.

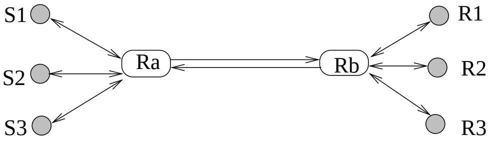

Fig. 2. Architecture of test-bed used for the experiments

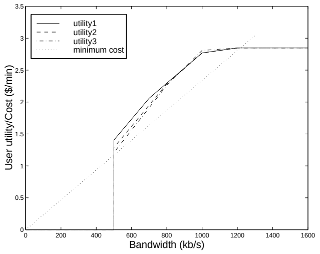

Fig. 3. Utility functions used in the experiments

The above algorithm is repeated for each possible combination of service-specific utility functions; each time, an optimal transmission parameter set is obtained. The transmission parameter set with the highest total surplus is then selected.

IV. Experimental Results and Analysis

We now describe a set of experiments which address the following issues: (i) the adaptive sharing of bandwidth between competing applications with identical utility functions, and the sharing of bandwidth between competing applications with utility functions reflecting different amounts of elasticity in bandwidth requirements; (ii) distribution of bandwidth so as to maximize mission-wide value; (iii) the influence of specific changes in the utility function on the bandwidth adaptation.

Experiments were carried out over a very simple topology, consisting of two routers connected by a 10 Mb/s link (Fig. 2). Three RNAP sessions were established end to end, and shared the same output interface of the link. To create different levels of network load, a simple data source model was used in each session to continuously send UDP packets. The packet generation rate was tunable to allow a user to adapt to any data rate it intended to send.

Out of the total capacity of 10 Mb/s, 4 Mb/s was configured for Controlled-Load (CL) service. In addition to the CL traffic, background traffic was also sent using best effort service. We assumed a service roughly as expensive (per unit bandwidth) as a telephone line. Assuming a charge of 10 c/min for a 64 kb/s service, the usage price was set as 2.6 c/Mb when the link was un-congested - when usage exceeded 70% of the CL capacity (2.8 Mb/s), an additional usage-sensitive congestion price was charged. Detailed descriptions of the formulation of service price according to traffic volume are given in [2].

Bandwidth sharing between competing applications:

In the first experiment, we show results for three competing applications, each with the utility function *utility1* in Fig. 3. Initially, in response to the initial price, each user determines that the optimal bandwidth is 1000 kb/s, therefore the total reservation of 3000 kb/s is higher than the link congestion threshold of 2800 kb/s. Fig. 4-a1 shows the price stabilizing after about 5 negotiation periods, and the variation with time of the total bandwidth reservation. Fig. 4-a2 shows the maximum per-user bandwidth that the user budget permits - as the affordable bandwidth decreases, each user is constrained to decrease its sending rate in response, and all users are observed to have nearly identical adaptation traces.

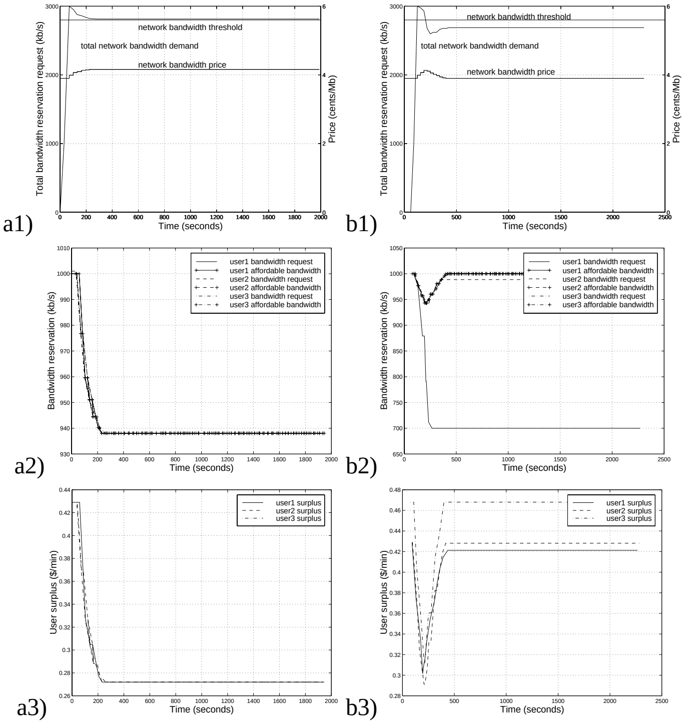

Fig. 4. Allocation of bandwidth and surplus for three competing users sharing a link. a1, a2, and a3 show the results when the users all have the utility 1 function from Fig. 3, and b1, b2, and b2 show corresponding results when the users have different utility functions from the same figure

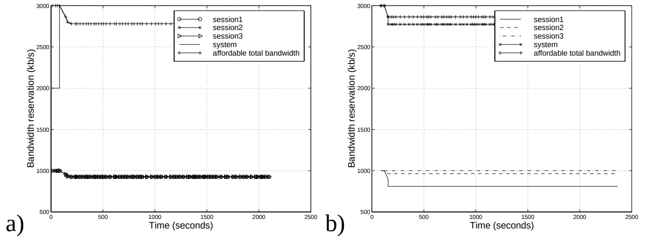

Fig. 5. Bandwidth reservation: a) all media have the same utility functions; b) All media sessions have different utility functions.

The second experiment is similar except that the three media sessions have different utility functions of Fig. 3. An utility function with a smaller slope reflects a higher elasticity in the bandwidth requirement of the user. Fig. 4-b2 shows that the user with the more elastic requirement is more sensitive to price changes and reduces his resource requirement faster when the network price increases, although all three users continue to obtain a similar perceived surplus (Fig. 4-b3). Thus, users with less stringent bandwidth requirements express this flexibility through a less bandwidth-sensitive utility function, and bear a greater share of reductions in bandwidth for congestion-control. Users with more bandwidth-sensitive requirements have to pay a higher charge during congestion to maintain their bandwidths at current levels.

Bandwidth sharing across a multimedia system:

In the third experiment, the distribution of available bandwidth across three component applications of a single multimedia system was studied. When all three applications had the *utility1* function, Fig. 5a shows that the total system bandwidth is equally distributed between them at all times. When the three applications had separately the three utility functions of Fig. 3, Fig. 5b shows that the media session with the more elastic resource demand is assigned relatively less bandwidth so as to maximize the overall perceived value across the system.

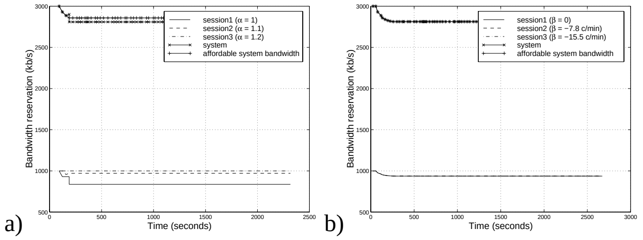

Fig. 6. Bandwidth reservation with utilities scaled multiplicatively a) and shifted additively b)

Effect of scaling and shifting utility function:

We also studied how the bandwidth adaptation is influenced by linear operations on the utility function - an offset applied uniformly to the utility over all bandwidths, and a multiplicative scaling of the function. Such linear operations could be used, for example, to reflect an evolution with time of the value of a particular information stream, or the evolution of relative importance of individual applications in a system.

A multiplicative scaling of the utility function by a factor greater than one tends to increase its bandwidth share since it reduces the demand elasticity of the application. Consider three media sessions belonging to a system with utility functions given by scaling *utility1* of Fig. 3 by a factor of 1, 1.1, and 1.2 respectively. Fig. 6a shows an application with a higher scaling factor gets a larger bandwidth under congestion.

Alternatively, a constant offset to the utility function will not influence the resource distribution as long as the valuation of a bandwidth is higher than its cost. This is because the utility function represents the relative preference of the user for different bandwidths. But it changes the minimum perceived value, which represents the user's willingness to pay to just keep the application alive. Fig. 6b shows that three sessions, with utility functions obtained by applying different offsets relative to the utility function *utility1*, are allocated the same bandwidth.

V. Conclusion

We have presented a framework for incentive-driven rate and QoS adaptation by an application or multi-application system. In this framework, the user responds actively to changes in price signaled by the network by dynamically adjusting network resource usage by the application. The adaptation is based on the user-perceived value of a given combination of transmission parameters, relative to the cost of obtaining the corresponding service from the network, taking into account constraints imposed by the minimum application requirements and the budget specified by the user. In a multi-application system such as a video-conference application, the system budget is distributed among the component media according to changes in price, as well as changes in the relative utilities with time or under different application scenarios, so as to maximize the overall perceived value relative to cost. Some heuristics are discussed to simplify this process. Experimental results show that perceived value based adaptation allows bandwidth to be shared among competing users or applications in a system fairly. At the onset of congestion, the bandwidth share of users with more elastic demands is reduced more, but all users receive equitable levels of perceived surplus. Multiplicative scaling and additive shifting of utility functions can be used to control the evolution of application utilities with time.

References

[1] X. Wang, H. Schulzrinne, "Comparison of adaptive Internet multimedia applications," *IEICE Transactions on Communications*, pp. 806-818, June, 1999.

[2] X. Wang and H. Schulzrinne, "RNAP: A Resource Negotiation and Pricing Protocol", *International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'99)*. Basking Ridge, New Jersey, pp. 77-93, Jun. 1999

[3] S. Shenker, C. Partridge, and R. Guerin, “Specification of guaranteed quality of service,” RFC 2212, Sept. 1997.

[4] J. Wroclawski, “Specification of the controlled load quality of service,” RFC 2211, Sept. 1997.

[5] V. Jacobson, K. Nichols, and K. Poduri, “An expedited forwarding PHB,” Internet Draft, Internet Engineering Task Force, Feb. 1999. Work in progress.

[6] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski, “Assured forwarding PHB group,” Internet Draft, Internet Engineering Task Force, Feb. 1999. Work in progress.