Fundamental tradeoffs in distributed algorithms for rate adaptive multimedia streams*

Vilas Veeraraghavan and Steven Weber

Drexel University, Department of ECE, Philadelphia, PA 19104
{vv35@drexel.edu,sweber@ece.drexel.edu}

Abstract. Rate adaptive multimedia streams are streaming media connections where the encoding rate is adjusted dynamically (with corresponding changes in media content resolution) in response to changing levels of congestion along the connection. The field of optimization based congestion control has yielded sophisticated distributed algorithms for resource allocation among competing elastic streams. In this work we study the fundamental tradeoffs for a class of optimization based distributed algorithms for rate adaptive streams, building on our earlier work. We focus on three tradeoffs: *i*) the tradeoff between maximizing client average quality of service (QoS) and client fairness, *ii*) the tradeoff between granularity of control (both temporal and spatial) and QoS, and *iii*) the tradeoff between maximizing the received volume and minimizing the fluctuations in received rate. These tradeoffs are illustrated through extensive simulation results using `ns-2`.

1 Introduction

1.1 Optimal congestion control for rate adaptive streams

Optimal congestion control for elastic traffic is a mathematically tractable optimization problem because of both the separable nature of the objective function (sum user utility) and the assumption that the individual user utility functions are strictly concave increasing. As pointed out by Shenker [1], the utility function for a rate adaptive stream will be concave increasing but will have a convex increasing neighborhood around zero, capturing the fact that even rate adaptive media has an associated minimum required rate for satisfactory service quality.

This convex neighborhood around zero complicates the mathematical analysis substantially. Recent work by Lee, Mazumdar, and Shroff [2] and Chiang, Zhang, and Hande [3] has thoroughly discussed solutions to optimization problems of this type. The work by Lee et al. uses the theory of subdifferentials to obtain traction on the non-convex optimization problem, although the possible existence of a duality gap limits the value of this approach in the absence of further restrictions. They propose the "self-regulating" property, which essentially requires users make reasonable allocation requests that ensure non-negative net utility (benefit minus costs).

* This work is supported by the NSF under grant 0435247.

The work by Chiang et al. takes a different focus by identifying necessary and sufficient conditions under which the distributed algorithms designed for concave utility functions converge to globally optimal resource allocations even when the actual utility functions are are not concave. Furthermore, the authors identify several practical issues that complicate deployment of distributed algorithms for rate adaptive streams: timescale and causality. In particular, they establish that prices need to be generated on a faster time scale than required for elastic traffic, and that optimal prices at time t may depend upon optimal prices at a *future* time $t' > t$.

The focus of [2, 3] is on the mathematical complexities associated with non-concave optimization problems and their corresponding distributed solutions. Our focus, instead, is on three fundamental tradeoffs not explicitly addressed in [2, 3]: optimality versus fairness, granularity of control, and received volume versus rate fluctuations (discussed in detail below).

1.2 Three fundamental tradeoffs

Our model of a rate adaptive stream is based on the notion of stream *volume* (v), defined as the product of the maximum desired bit rate at full resolution ($s^{\max}$) times the stream duration (d). For simplicity, we assume the maximum desired bit rate is a constant, even though in the common case of VBR encoding of video the maximum rate will be time-varying. Thus $v^{\max} = s^{\max}d$ is the total number of bits associated with the stream. By employing dynamic rate adaptation, corresponding to a time-varying instantaneous received rate $s(t) \leq s^{\max}$, the client will receive a total number of bits $v = \int_0^d s(t)dt$.

Quality of service. We study two distinct quality of service measures, the time average utility and the rate of adaptation, which serve as a proxy for client-perceived media quality. The first metric, q, is the time average (over the stream duration) of the instantaneous utility, $u(x)$, where instantaneous utility is measured as a function of the normalized received rate $s(t)/s^{\max}$. Note that in the case where $u(x) = x$ (linear utility), we have $q = v/v^{\max}$, the fraction of bits received. The *rate of adaptation*, r, is simply the time average rate of change of the instantaneous received rate, i.e., the sum of the magnitudes of the changes in rate divided by the duration. We have selected these two QoS metrics because they each capture an important part of the overall rate adaptive streaming media client experience. In particular, q captures the fact that higher instantaneous rates yield higher instantaneous utility, while r captures the fact that fluctuations in the encoding level are both aurally and visually distracting.

Tradeoff #1: optimality and fairness. We assume the primary objective of the network is to maximize the client average time average utility. Since each stream is counted equally in computing the client average, it follows that the client average is maximized by giving preferential treatment to small volume users. Intuitively, allocating resources to small volume streams goes further proportionally to improving their QoS than does allocating resources to large volume streams. We term a resource allocation policy that attempts to maximize

client average QoS by giving preferential treatment to small clients a *volume discriminatory policy*. It is clear that, although small volume streams are satisfied under such an allocation, users requesting large volume streams (say, of high rate streaming video content) will bristle at the policy. Fundamentally the tension can be seen as that between optimality and fairness.

Tradeoff #2: granularity of control and QoS. Dynamics rate adaptation can be done in real time by on the fly re-encoding of the media content to yield the desired bit rate, or "stored" by selecting among one of a discrete set of available encodings. The tradeoff is roughly this: real-time encoding is computationally burdensome and therefore infeasible for large scale media servers, but has the singular benefit that the transmission rate can be tuned at an arbitrarily fine granularity of control. Stored encoding is more scalable computationally (although storage may become an issue), but the available transmission rates are limited to the stored encodings. We call the number of encodings the *spatial granularity of control*.

A second natural tradeoff for any distributed algorithm is that between the optimality of the resource allocation obtained by the algorithm and the time between state updates. Clearly the performance of the algorithm improves in the timeliness of the feedback. This improvement may slow for very rapid state updates as the time scale of the updates becomes much faster than the time scale of the changes in system state. We call the time between updates the *temporal granularity of control*.

Tradeoff #3: time average utility and rate of adaptation. There is an inherent tension between the objective of maximizing the time average utility (q), and the objective of minimizing the rate of adaptation (r). As mentioned above, both q and r are important in that q captures the notion that higher resolution encodings yield higher user satisfaction, while r captures the notion that changes in encoding level detract from the user experience. It is intuitively clear that the highest received rate is obtained by an algorithm that is capable of instantly adjusting the encoding level to match the available capacity on the link. Such an algorithm will clearly maximize q, but in an environment where the available capacity is changing rapidly, will also incur a possibly unacceptably high rate of adaptation, r. Of course minimizing r is easy: simply avoid dynamic rate adaptation altogether, which has the cost of foregoing a significantly higher average received rate obtainable through dynamic adaptation.

2 Controllers for rate adaptive streams

The controllers presented in this section are developed in our earlier work [4] (that work does not discuss the three tradeoffs which form the heart of this work). The controllers are similar to but distinct from the controllers developed by [2, 3]. The first distinction is that we assume an admission control mechanism limits the traffic on the link such that each stream is assured of receiving its minimum granularity encoding, and that the utility function is concave for encodings above the associated minimum rate; this removes the focus on the

convex neighborhood around zero which is central to [2, 3]. Second, our algorithm is designed to maximize our primary QoS metric, the *time average utility* q, whereas the algorithms in [2, 3] are designed to maximize the *instantaneous utility*. Third, we emphasize the use of a near-optimal *discrete* controller, which selects the encoding level among the discrete set of available encodings, whereas [2, 3] focus on continuous controllers.

2.1 Network model, stream model, and QoS metrics

Network model. We let $\mathcal{L}$ denote the set of links in the network, and the vector $\mathbf{c} = (c_l, l \in \mathcal{L})$ denote the capacities of those links. We assume that the streaming traffic is given priority over the best-effort traffic on the network, so that the entire link capacity is available to the streaming traffic. We recognize this is a major assumption, but our focus in this work is not on observing the co-existence of streaming and elastic traffic in the model. Each client-server pair is identified with a unique and fixed route through the network. Let $\mathcal{R}$ denote the set of routes, where a route r is composed of a set of links $\{l \in r\} = \{l \mid l \in r\}$. The vector $\boldsymbol{\lambda} = (\lambda_r, r \in \mathcal{R})$ denotes the arrival rate of new stream requests on each route. We index the admitted streams on each route, so that (i, r) denotes the i^{th} admitted stream on route r.

Stream model. We model a rate adaptive stream by five quantities: i) stream duration (d), ii) minimum subscription level ($s^{\min}$), iii) maximum subscription level ($s^{\max}$), iv) the instantaneous normalized rate utility function $u : [0, 1] \to [0, 1]$, and v) the weight (w), reflecting the relative importance of the stream. All five quantities will in general be stream-dependent.

Each stream (i, r) has its individual minimum and maximum subscription level denoted by $(s^{\min}_{i,r}, s^{\max}_{i,r})$. We assume the utility function for each client, $u_{i,r} : [0, 1] \to [0, 1]$, is a twice differentiable strictly concave increasing function with a convex neighborhood around zero. The argument of the utility function is the fractional rate received, i.e., if the client receives rate $s_{i,r}$ then the utility is $u_{i,r}(s_{i,r}/s^{\max}_{i,r})$. We *define* $s^{\min}_{i,r}$ as the rate where the utility function switches from convex to concave. We consider both continuous and discrete controllers. A continuous controller is capable of creating an encoding of any desired rate "on the fly" $s_{i,r} \in [s^{\min}_{i,r}, s^{\max}_{i,r}]$. A discrete controller is capable of using any of a set $\mathcal{S}_{ir} = \{s^{\min}_{i,r}, \ldots, s^{\max}_{i,r}\}$ "stored" encodings.

The admission control rule is this: *admit a new stream i on route r as long as there is sufficient capacity to satisfy the minimum rate requirements of the previously admitted streams as well as that of the stream seeking admission:*

$$s^{\min}_{i,r} + \sum_{r' \ni l} \sum_{j=1}^{n_{r'}} s^{\min}_{j,r'} \leq c_l, \ l \in r, \tag{1}$$

where n_r is the number of active streams on route r at the time of request. Note that the admission process is completely separate from the allocation process.

Quality of service metrics. The first quality of service metric is the time average utility:

$$q_{i,r} = \frac{w_{i,r}}{d_{i,r}} \int_{a_{i,r}}^{a_{i,r}+d_{i,r}} u(s_{i,r}(t)/s_{i,r}^{\max})dt, \tag{2}$$

where $a_{i,r}$ is the admission time, $d_{i,r}$ is the duration, and $w_{i,r}$ is an assigned weight. The second metric is the rate of adaptation:

$$r_{i,r} = \frac{1}{d_{i,r}} \int_{a_{i,r}}^{a_{i,r}+d_{i,r}} \Big| s_{i,r}(t) - s_{i,r}(t^+) \Big| dt. \tag{3}$$

Note that r is used both to indicate a route and the rate of adaptation; the meaning will be clear from context. We let q be the *primary* QoS metric, and r be the *secondary* QoS metric. Thus when we speak of maximizing QoS we will always mean maximizing q.

2.2 Continuous rate controller

In our earlier work [4] we show that the the weighted client average QoS is maximized provided the resource allocation at each point in time t is the solution of a weighted sum utility optimization problem:

$$\max_{\mathbf{s}\in\mathcal{S}} \Big\{ \sum_{(i,r)\in\mathcal{N}(t)} \frac{w_{i,r}}{d_{i,r}} u_{i,r}\Big(\frac{s_{i,r}}{s_{i,r}^{\max}}\Big) \;\Big|\; \sum_{r\ni l} \sum_{i\in\mathcal{N}_r} s_{i,r} \le c_l,\ l \in \mathcal{L} \Big\}. \tag{4}$$

where $\mathcal{S}$ is set of all feasible allocations for active streams, $\mathcal{N}(t)$ is the set of active streams at time t and $\mathcal{N}_r$ is the set of active streams on route r. This objective plays the same role as the *SYSTEM* problem originally formulated by Kelly in [5]. Recall our assumption that the optimization must ensure each stream receives its minimum encoding rate or higher. This assumption, and the definition of the minimum rate as the point at which the utility function switches from convex to concave, allow us to apply Kelly's distributed algorithm framework in [6] to the above problem. The resulting controller is:

$$\dot{s}_{i,r}(t) = \kappa s_{i,r}(t)\Big(\frac{w_{i,r}}{v_{i,r}} u'_{i,r}\Big(\frac{s_{i,r}(t)}{s_{i,r}^{\max}}\Big) - p_r(t)\Big), \quad (i,r) \in \mathcal{N}(t), \tag{5}$$

As mentioned in the introduction, this controller is of the same canonical form as that proposed by Kelly et al. in [6], where κ is the gain constant and $p_r(t)$ is the route price, assumed to be additive over the instantaneous link costs comprising the route. The only difference from Kelly's formulation is that we require that the continuous controller maintain a rate in the interval $[s_{i,r}^{\min}, s_{i,r}^{\max}]$. Thus we set $\dot{s}_{i,r}(t) = 0$ if either the route price $p_r(t)$ is high and $s_{i,r}(t) = s_{i,r}^{\min}$ or the route price is low and $s_{i,r}(t) = s_{i,r}^{\max}$.

The *volume dependent* continuous controller sets each $w_{i,r} = w$, while the *volume independent* controller sets each $w_{i,r} = v_{i,r}$. To encourage fair comparison between these two controllers we select the weight w such that $\mathbb{E}\Big[\frac{w}{V}\Big] = 1$, where the expectation is taken with respect to the distribution of the volume of the the admitted streams.

2.3 Discrete rate controller

Introduced in [4], our proposed discrete controller works as follows. Each stream runs a *virtual controller* that computes $\dot{s}^{\text{vir}}_{i,r}(t)$ from the continuous controller of the previous section, and from this computes $s^{\text{vir}}_{i,r}(t)$ in response to the updated route prices $p_r(t)$. Each stream employs a pair of thresholds $(z^{\min}_{i,r},\ z^{\max}_{i,r})$ such that

$$s^{\min}_{i,r} < z^{\min}_{i,r} \leq \frac{s^{\min}_{i,r} + s^{\max}_{i,r}}{2} \leq z^{\max}_{i,r} < s^{\max}_{i,r}. \tag{6}$$

The subscription level changes according to the following rule:

$$\begin{aligned} s^{\min}_{i,r} &\Rightarrow s^{\max}_{i,r} \quad \text{if} \ \ s^{\text{vir}}_{i,r}(t) > z^{\max}_{i,r}, \\ s^{\max}_{i,r} &\Rightarrow s^{\min}_{i,r} \quad \text{if} \ \ s^{\text{vir}}_{i,r}(t) < z^{\min}_{i,r}. \end{aligned} \tag{7}$$

Note that when we set

$$z^{\min}_{i,r} = \frac{s^{\min}_{i,r} + s^{\max}_{i,r}}{2} = z^{\max}_{i,r}, \tag{8}$$

the above algorithm simply selects available subscription level nearest to the virtual subscription level. Setting the min and max thresholds strictly below and above the median subscription level serves to retard the frequency of subscription level changes. In particular, if a stream is low then the virtual subscription level must actually rise above $z^{\max}_{i,r}$ to induce an increase. Similarly, if a stream is high then the virtual subscription level must drop below $z^{\min}_{i,r}$ to induce a decrease. This serves as a hysteresis mechanism to retard the fluctuations in subscription level which have an adverse effect on the rate of adaptation metric.

3 Three fundamental tradeoffs

We study the three fundamental tradeoffs in distributed algorithms for rate adaptive streams. We present simulation results obtained by implementing the continuous and discrete controllers from the previous section in `ns-2`. Due to space constraints we restrict our attention in this section to a single link model of capacity c (kbps).

User utility function. Similar to our example in [4], we presume that all streams employ a common (sigmoid) utility function:

$$u(x) = \frac{1}{1 + \rho e^{-\sigma(x-\gamma)}}, \quad \rho > 0, \quad \sigma > 0, \quad \gamma \in (0,1). \tag{9}$$

Recall that the argument of the utility function is the fractional rate $x = s/s^{\max}$. Thus u has a convex neighborhood around zero extending to $x = \gamma$, and is then concave for $x > \gamma$. The parameter σ governs the shape. The parameter ρ governs the height of the function for x near γ. We have selected $\gamma = \frac{1}{2}$, $\rho = 3$, and $\sigma = 10$. This means each stream has a minimum rate that is 50% of its

maximum rate, i.e., $s^{\min}/s^{\max} = 1/2$. Moreover, the minimum rate is presumed to account for approximately 50% of the possible quality or satisfaction, since $u(s^{\min}/s^{\max}) = u(\gamma) = 1/(1+\rho) = 1/2$, while $u(1) \approx 1$.

Maximum subscription level and stream duration. Streaming media content varies widely in both the maximum subscription level (small bit rates for audio content, high bit rates for HD video content), and the duration (short durations for songs, long durations for movies). To capture this diversity in the simplest manner possible, we employ an elephants and mice model where both the maximum subscription level, $S_{i,r}^{\max}$ and the stream duration, $D_{i,r}$, are Bernoulli random variables. The notation $X \sim \text{Ber}(s, l, p)$ denotes a random variable X is Bernoulli with $p = \mathbb{P}(X = s) = 1 - \mathbb{P}(X = l)$, where we think of s for small and l for large. Define the constants

$$\begin{array}{cccccccc} \hat{s}^{\max,\min} & \hat{s}^{\max,\max} & p_s & \sigma & \hat{d}^{\min} & \hat{d}^{\max} & p_d & \delta \\ 128 & 1280 & 0.5 & 704 & 60 & 600 & 0.5 & 330 \end{array} \tag{10}$$

All rates are in kbps and all durations are in seconds. Define the volume diversity parameter $a \in [0, 1]$ and the diversity spread functions

$$\begin{aligned} s^{\max,\min}(a) &= (1-a)\sigma + a\hat{s}^{\min,\max} \\ s^{\max,\max}(a) &= (1-a)\sigma + a\hat{s}^{\max,\max} \\ d^{\min}(a) &= (1-a)\delta + a\hat{d}^{\min} \\ d^{\max}(a) &= (1-a)\delta + a\hat{d}^{\max}. \end{aligned}$$

For fixed a, we set

$$\begin{aligned} S_{i,r}^{\max}(a) &\sim \text{Ber}(s^{\max,\min}(a), s^{\max,\max}(a), p_s) \\ D_{i,r}(a) &\sim \text{Ber}(d^{\min}(a), d^{\max}(a), p_d). \end{aligned}$$

First observe that the volume diversity parameter does not affect the mean for either S or D:

$$\mathbb{E}[S_{i,r}^{\max}(a)] = \sigma, \quad \mathbb{E}[D(a)] = \delta, \quad a \in [0, 1]. \tag{11}$$

Note that for $a = 0$ the Bernoulli values coincide, i.e., $s^{\max,\min}(0) = s^{\max,\max}(0) = \sigma$ and $d^{\min}(0) = d^{\max}(0) = \delta$, while for $a = 1$ the Bernoulli values take on their extreme values: $s^{\max,\min}(1) = \hat{s}^{\max,\min}, s^{\max,\max}(1) = \hat{s}^{\max,\max}$ and $d^{\min}(1) = \hat{d}^{\min}, d^{\max}(1) = \hat{d}^{\max}$. Thus increasing a from 0 to 1 increases the diversity of stream volumes found on the link while not affecting the mean volume $\mathbb{E}[V] = \sigma\delta$. Recall that for the volume dependent algorithm we select w such that $\mathbb{E}[1/V] = w$; for the current model this yields $w = (400/121)\sigma\delta$. Finally, for the discrete controller our default selection (aside from Figure 2) is to use $K = 2$ encodings: $\mathcal{S} = \{s^{\min}, s^{\max}\}$.

Link capacity and loading. We will devote significant attention to studying the QoS under the controllers as the link capacity is varied (while the arrival rate is held constant). Note that when the capacity per stream is near or smaller than the average minimum required rate per stream, that the typical stream will

spend most of its tenure at its minimum rate. If the capacity per stream is at or exceeds the average maximum rate per stream then the typical stream will spend most of its time at its maximum rate. With this in mind we parameterize the link capacity as $c = m\lambda\sigma\delta$, where λ is the arrival rate (assumed Poisson). Note that, barring blocking, the average number of streams on the link is $\mathbb{E}[|\mathcal{N}_t|] = \lambda\delta$ (by Little's Law), and as such the capacity per stream is $c/(\lambda\delta) = m\sigma$. At $m = 1$ the capacity per stream matches the typical maximum subscription level. Recalling that the user utility function sets $s^{\min}/s^{\max} = 1/2$, we see that for $m < 1/2$ the capacity per stream matches the typical minimum subscription level. Following [7], we term the regime $m \in [0, 1/2]$ the *overloaded regime*, $m \in [1/2, 1]$ the *rate adaptive regime*, and $m \in [1, \infty)$ the *underloaded regime*. We have selected λ so that on average (barring blocking) there are 10 streams sharing the link, thus $\lambda = 10/\delta$. It follows that the link capacity at the scaling threshold $m = 1/2$ is $c = m(\lambda\delta)\sigma = 1/2 \cdot 10 \cdot 704 = 3520$, while and at $m = 1$ is 7040. In our capacity plots we will vary m from 0.1 to 1.2.

When the link capacity is fixed, we will use $m = 1/2$, which corresponds to provisioning the link the rate adaptive regime such that the capacity per stream is twice what is minimally required by a typical stream, but half of the maximum rate requested by a typical stream.

Simulation setup. We have implemented our controllers in `ns-2` [8] which provides ample support and a realistic simulation environment to test our model. It ensures us results that we are most likely to see during an actual real-world implementation. We set up 1000 nodes acting as transmitters connected by a single bottleneck link to 1000 receiver nodes. The bottleneck link has a packet queue of size 100 used to smooth the traffic and implement a DropTail policy for the packets. This link is a duplex link allowing acknowledgment packets from the receivers to reach the transmitter indicating if the packet has been lost or received without incident. All the transmitting nodes are UDP sources and the receivers are Loss Monitoring agents. We use a CBR (Constant Bit-Rate) traffic pattern for each node. The values for the parameters of each stream like duration and subscription levels are assigned as discussed previously. Each simulation point is averaged over 1000 streams and over 100 repetitions of the experiment.

3.1 Tradeoff #1: Optimality and fairness

The first tradeoff we study is between optimality and fairness. In particular, the volume dependent controller optimizes the client average quality of service by giving preference to small volume streams, while the volume independent controller treats all streams fairly. The top plot in Figure 1 presents QoS for both controllers as the link capacity is increased, while the bottom plot presents QoS for both controllers as the volume diversity parameter is increased. For the top plot the volume diversity parameter is maximized at $a = 1.0$. For the bottom plot the link capacity is selected using a capacity scaling parameter of 0.5. The plots illustrate how the volume dependent controller is able to exploit volume diversity to maximize the client average QoS, with pronounced improvements

in the rate adaptive capacity scaling regime, and when the volume diversity parameter is large.

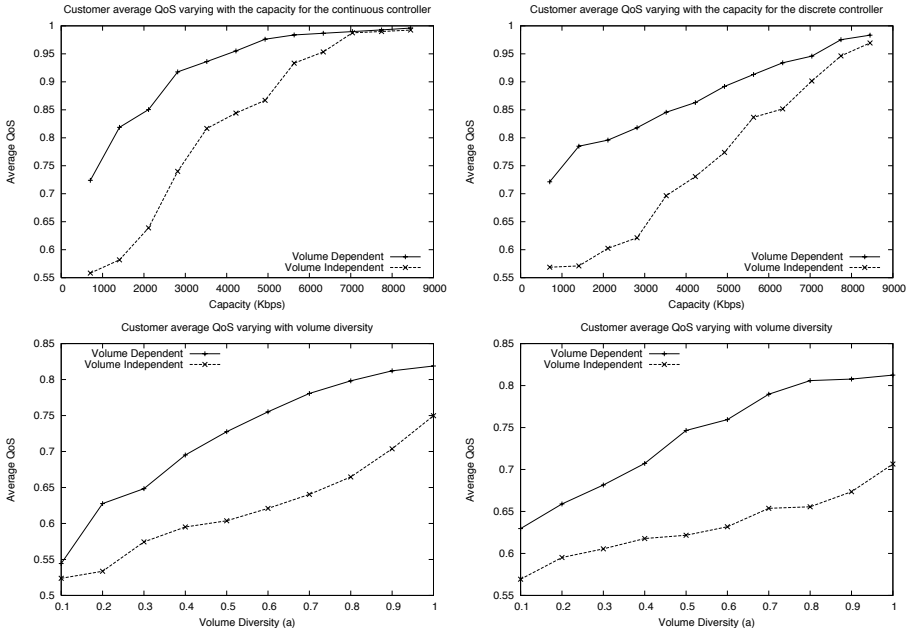

Fig. 1. Tradeoff #1: comparing optimality and fairness. All four plots compare the QoS performance of the volume dependent controller with that of the volume independent controller. The **top two** plots vary the link capacity, c (holding $a = 1.0$), and illustrate the significant increase in QoS achieved by the volume dependent controller, especially in the overloaded and rate adaptive regimes. The **bottom two** plots vary the volume diversity parameter, a (holding $m = 0.5$), and show that the improvement in QoS achievable by the volume dependent controller increases in a. In both cases we present results for both continuous and discrete controllers (using `ns-2`).

3.2 Tradeoff #2: Granularity of control and QoS

We next consider the impact of both temporal and spatial granularity of control on the QoS. Temporal granularity refers to the time between state updates, while spatial granularity refers to the number of available subscription levels. The top plot in Figure 2 shows the loss rate as the time between updates is increased; as expected the loss rate increases. More surprising is that the volume dependent controller achieves a significantly lower loss rate than the volume dependent controller. This can be explained by the fact that the volume dependent controller gives preference to smaller rate streams, which make proportionally smaller changes in rate, and as such they "feel out" the available capacity more

gradually than the large volume streams. The middle plot in Figure 2 shows the impact on the QoS as the time between state updates is varied from $\tau = 5$ to $\tau = 60$ (we used $\tau = 1$ for the plots in Figure 1). There is a significant impact on both controllers, but no "cliff", meaning the controller performance degrades gracefully in the absence of updates. The bottom plot in the figure shows the QoS as the link capacity is scaled (again from $m = 0.1$ to $m = 1.2$) with a varying number of encodings, K, available. The K encodings are assumed to be uniformly spaced over the interval $[s_{i,r}^{\min}, s_{i,r}^{\max}]$. Note that the curve $K = \infty$ is the continuous controller. Clearly there is a law of diminishing returns as K increases, and the content provider would be able to assess the tradeoff between the cost of storing more encodings with the marginal benefit to client QoS.

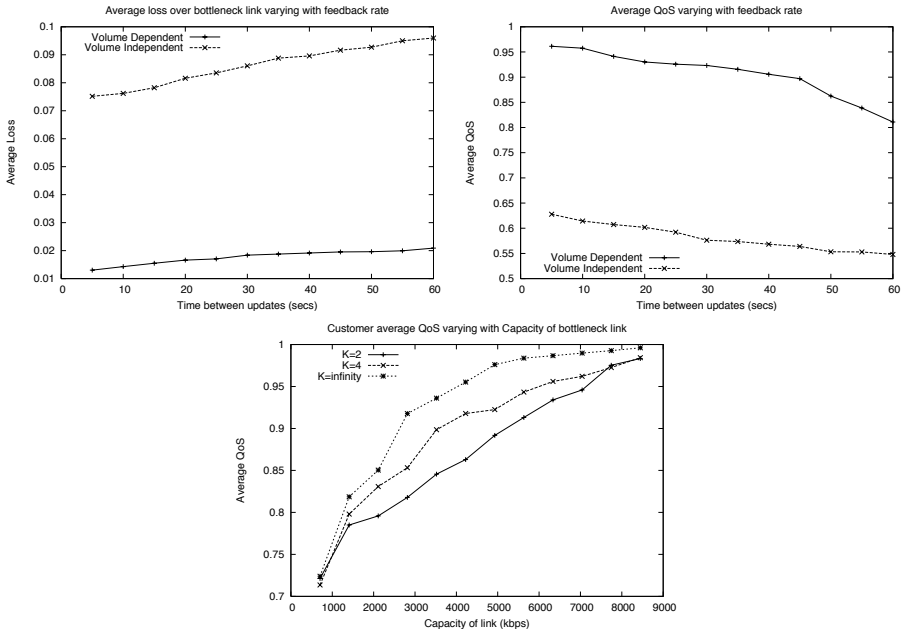

Fig. 2. Tradeoff #2: the impact of the granularity of control on the QoS. The **top** plot shows the loss rate as a function of the time between updates for both the volume dependent and volume independent controller. As expected the loss increases as time between updates increases, more surprising is the fact that the volume independent controller suffers significantly higher loss than the volume independent controller. The **middle** plot shows the impact of increasing the time between state updates on the QoS for both the volume dependent and volume independent controllers (both the continuous controller with $m = 0.5$). The **bottom** plot shows the impact of varying the number of offered encodings as the link capacity is scaled (holding $a = 1.0$). All plots are from `ns-2`.

3.3 Tradeoff #3: Time average utility and rate of adaptation

The third tradeoff is that between the competing aims of maximizing the time average utility (q) and minimizing the rate of adaptation (r). The top two plots in Figure 3 shows the change in q and r as the weight w is increased. There is a clear dependence of q, r on w for the continuous controller, and none for the discrete controller. This is because the discrete controller's update rule is relatively insensitive to the weight, depending only on the mapping between the virtual controller and the available set of discrete rates. The bottom plot shows the inherent tradeoff between maximizing q and minimizing r: the x-axis is r and the y-axis is q, the points are the QoS pairs $(q(w), r(w))$ as the weight w is increased from $w = 1$ to $w = 500$. The continuous controller shows a significant increase in q, but at the cost of an increase in r; the discrete controller (with $K = 2$ encodings) is again less sensitive to the weight, but also unable to fully achieve the q levels obtainable by the continuous controller.

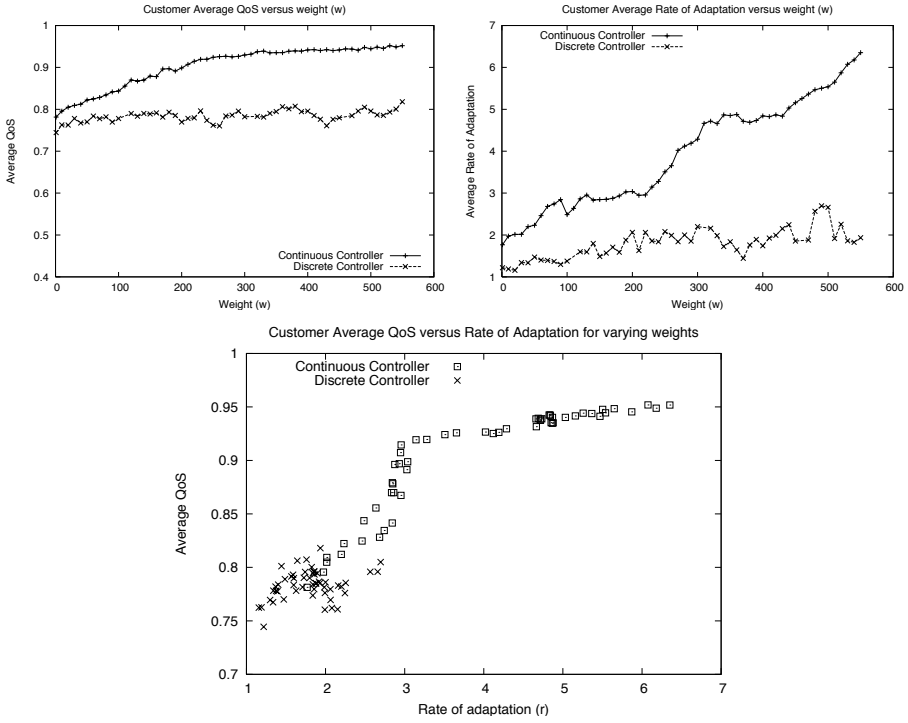

Fig. 3. Tradeoff #3: The tradeoff between maximizing the time average utility and minimizing the rate of adaptation. The **top** plot shows the increase in QoS obtained by increasing the weight w for both the continuous and discrete controller. The **middle** plot shows the increase in the rate of adaptation as the weight of the controller is increased (for $m = 0.5, a = 1.0$), and the **bottom plot** is a scatterplot of $(r(w), q(w))$ as the weight is increased from $w = 1$ to $w = 500$.

4 Conclusion

This paper has focused on three fundamental design tradeoffs in designing distributed algorithms for rate adaptive multimedia streams: *i*) the tradeoff between optimality and fairness, *ii*) the tradeoff between granularity of control and QoS, and *iii*) the tradeoff between maximizing the time average utility and minimizing the rate of adaptation. Although each of these tradeoffs are qualitatively intuitive, the quantitative results are instructive, and offer structural insights into the sometimes complex dependencies among the system parameters.

References

1. Shenker, S.: Fundamental design issues for the future internet. IEEE JSAC **13**(7) (September 1995)
2. Lee, J.W., Mazumdar, R.R., Shroff, N.B.: Non-convex optimization and rate control for multi-class services in the Internet. IEEE/ACM Transactions on Networking **13**(4) (August 2005) 827–840
3. Chiang, M., Zhang, S., Hande, P.: Distributed rate allocation for inelastic flows: optimization frameworks, optimality conditions, and optimal algorithms. In: Proceedings of IEEE INFOCOM, Miami, FL (March 2005)
4. Weber, S., Veeraraghavan, V.: Distributed algorithms for rate-adaptive media streams. Springer Networks and Spatial Economics (submitted) (May 2006)
5. Kelly, F.: Charging and rate control for elastic traffic. European Transactions on Communications **8** (1997) 33–37
6. Kelly, F., Maulloo, A., Tan, D.: Rate control in communication networks: shadow prices, proportional fairness, and stability. Journal of the Operational Research Society **49** (1998) 237–252
7. Weber, S., de Veciana, G.: Rate adaptive multimedia streams: optimization and admisssion control. IEEE/ACM Transactions on Networking **13**(6) (December 2005) 1275–1288
8. Network Simulator `ns-2`, `http://www.isi.edu/nsnam/ns/`