Optimization-Based Congestion Control for Multicast Communications

Jonathan K. Shapiro, Don Towsley, and Jim Kurose, University of Massachusetts at Amherst

ABSTRACT

This article outlines an approach for multicast congestion control based on an economic model that has been successfully applied to unicast congestion control. In this model, congestion signals are interpreted as prices and congestion-controlled sessions as utility maximizing agents. A naive extension of the unicast model fails to achieve a reasonable balance between providing the incentives necessary to promote the use of multicast and ensuring that multicast sessions do not interact too aggressively with unicast sessions. We extend the model by introducing a rational definition of multicast utility. The revised model provides a basis for multicast congestion control protocols that provide incentives to use multicast but are necessarily unfair to unicast traffic. We show, however, that the degree of unfairness can be controlled by appropriately setting a design parameter with a limiting case of strict fairness.

INTRODUCTION

Widespread deployment of multicast communication in the Internet depends critically on the existence of practical congestion control mechanisms that allow multicast and unicast traffic to share network resources fairly. Most service providers recognize multicast as an essential service to support a range of emerging network applications including audio and video broadcasting, bulk data delivery, and teleconferencing. Nevertheless, network operators have been reluctant to enable multicast delivery in their networks, often citing concerns about the congestion such traffic may introduce [1]. The basic conflict is this: It is desirable to encourage use of multicast where appropriate to reduce the overall bandwidth demand of applications that transmit high-bandwidth data to many receivers, but the introduction of multicast sessions into the network must not deteriorate the performance of existing unicast traffic. The specific worry of operators is that multicast congestion control protocols may interact too aggressively with the standard unicast congestion control mechanism of TCP,[1] leaving unicast sessions starved for bandwidth. There is a clear need for multicast congestion control algorithms that are provably fair to unicast traffic if this concern is to be addressed.

Our aim in this article is to provide insight into the problem of multicast congestion control by describing a promising new approach for congestion control of single-rate multicast traffic. Our approach is based on an economic theory of resource allocation. The underlying theory is grounded in a well-studied mathematical framework, which allows us to reason about the fairness of the bandwidth allocation resulting from the interaction of congestion-controlled sessions. We argue that congestion control schemes with a slight bias in favor of multicast are necessary to provide the appropriate incentives to use the multicast service. Therefore, our mechanism is not strictly fair to unicast traffic, but can be tuned to provide an acceptable level of such "controlled unfairness."

BACKGROUND

MULTICAST

Multicast is a service that delivers packets from a sender to a group of receivers. IP Multicast (the network-level multicast standard in the Internet) provides this service by transmitting a copy of each offered packet once on each link of a tree with its root at the sender and a leaf at each receiver. The links in this tree are shared by other traffic, and some of these links may be congested. Multicast congestion control protocols are designed to detect this congestion and adapt the session's flow rate on each link to the available bandwidth. Well behaved protocols must also ensure that bandwidth on bottleneck links is shared fairly with other competing flows. So-called *single-rate* multicast protocols require that a given session use the same transmission rate on every link. Thus, the rate for the entire session is limited by the most bandwidth-constrained receiver. In single-rate protocols, rate adaptation is performed by the sender in response to receiver feedback. Single-rate multi-

This work was supported in part by the National Science Foundation (NSF) under grant numbers EIA -0080119 and ANI-9980552.

[1] TCP, which stands for Transmission Control Protocol, is the Internet standard unicast transport protocol. It includes mechanisms for both reliable data delivery and congestion control.

 0163-6804/02/$17.00

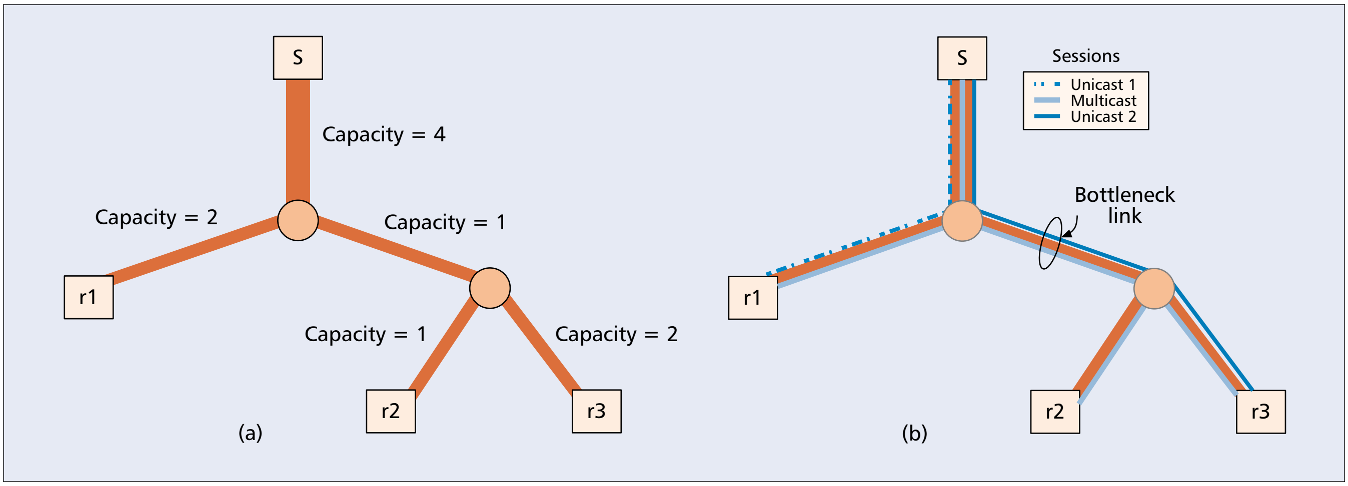


Figure 1. *a) A single sender (s) and three receivers (r1, r2, r3) inhabit this simple network topology; b) one multicast and two unicast sessions using the network. We will pay particular attention to how bandwidth is shared on the indicated bottleneck link.*

cast is widely used in real-world multicast applications. In contrast, so-called *multirate* protocols allow the session's flow rate on each link to vary depending on the bandwidth available to downstream receivers. This flexibility is achieved at the expense of degrading the data stream along bandwidth-constrained paths and is thus only appropriate for applications that use highly loss-tolerant data encoding techniques. Rate adaptation in multirate protocols is performed by receivers with support from the network. In this article we focus on single-rate multicast.

It is traditionally held that processing feedback from all receivers leads to unscalable multicast protocols. Protocol designs have therefore employed various ad hoc mechanisms to limit the set of receivers sending congestion signals to the sender. However, the interaction of such protocols with the standard TCP congestion control mechanism (and consequently their fairness properties) has been extremely difficult to analyze. Two recent developments have made it reasonable to consider designs that require feedback from all receivers. The first development is the emergence of proposed router-based services for aggregating feedback, which would allow a sender to process feedback (suitably summarized by the network) from the entire receiver set (e.g., [2]). The second is the development of a mathematical framework for understanding the effect of unicast congestion control protocols on the global network state. This framework borrows from mathematical models used by economists to study resource allocation in markets with self-interested participants. Such models are useful for studying congestion control, which is fundamentally a resource allocation problem in which bandwidth is the resource. Like players in a market, individual sessions in the network have a localized view of the entire system and, based on their individual application performance objectives, exhibit self-interested behavior. Issues of fairness are readily analyzed in this framework, but fair protocols require information about every link used by the session. In multicast, this requirement implies that the sender must collect feedback from all receivers.

Optimization-Based Congestion Control

Optimization-based congestion control employs a simple network model augmented with "economic" features. Consider a network modeled as a set of directed links with fixed but not, generally, equal capacities. The workload for the network is generated by a set of sessions. A session is described by the subset of network links over which it transmits data and by a variable transmission rate, denoted x. The aggregate data rate on any link, then, is the sum of rates for all sessions using that link. Each session is characterized by a *utility function* $u(x)$, which represents the value of bandwidth to the session. Utility functions have two important properties that capture natural intuitions about session behavior. First, utility is an increasing function of transmission rate because we assume that each session would prefer as high a rate as possible. Second, the utility function has decreasing marginal returns, which models the idea that the value of a small increase in transmission rate is high for a session currently transmitting at a low rate, but decreases as the session's rate increases. Two widely used utility functions in congestion control models are *logarithmic utility*,

$$u(x) = \log x,$$

and a utility function that is a good model for the behavior of TCP at low loss rates [3], which we will call *TCP utility*,

$$u(x) = -1/x.$$

Associated with each link is a real value, which, for reasons that will become clear shortly, we informally interpret as a "price" per unit bandwidth and refer to as the *link price*. For each session, then, we may define the *session price* as the sum of prices over all of the links it uses. Figure 1 shows a simple network with one sender and three receivers. As a running example, we will consider how bandwidth is allocated among one multicast session and two unicast sessions. Since the rates of the multicast session and unicast session 2 are constrained by the indicated bottleneck link, we will compare the bandwidth allocation on that link under several alternative schemes.

In designing congestion control protocols, we are essentially looking for a distributed algorithm whose rate setting behavior can be naturally interpreted as a decentralized iterative solution to this problem.

In this model, the problem of congestion control can be cast as one of utility maximization. From a networkwide perspective, it is desirable to maximize the *aggregate utility* of all sessions. To make this notion precise, we define aggregate utility as the sum of session utilities, allowing a precise statement of the congestion control problem.

Congestion Optimization Problem: *Find a set of rates for all sessions to maximize*

$$\text{aggregate utility} = \sum_{\text{all sessions } i} u(x_i)$$

subject to the constraint that the traffic on each link may not exceed its capacity.

In designing congestion control protocols, we are essentially looking for a distributed algorithm whose rate setting behavior can be naturally interpreted as a decentralized iterative solution to this problem. For such algorithms to be practical, it is essential that each individual session need only rely on locally available information to set its own rate. We therefore assume that each session behaves greedily by setting its rate to maximize its own individual utility minus the total price assessed by the network. (Recall that the session price is a charge per unit bandwidth, so the total price is the session price multiplied by the session's transmission rate.) It is worth emphasizing that despite suggestive names for quantities such as price in the model, one needn't think of the link prices as representing actual monetary charges. Rather, these quantities can be thought of as congestion signals, which play the role of aligning supply and demand, similar to the way prices behave in economic models of markets. The rate chosen by a session depends on the session price in an intuitive way: the lower the price, the higher the rate. The key point of optimization-based congestion control is that if the network sets the link prices to accurately reflect the current level of congestion, the sessions will converge to rates that jointly maximize the sum of their utilities, thereby optimizing the networkwide objective subject to the capacity constraints.

Economics Background: Prices, Utility and Social Welfare

Since economic models such as the one presented here may be unfamiliar to many people working in networks and communications, we next provide some background on the notions of prices, utility functions, and the theory of social welfare.

Although the concept of utility as a measure of satisfaction seems fairly intuitive to many, it can be difficult to quantify precisely. There is no clear unit of utility and no agreed on scale. Comparing the utility of two individuals can be tricky, particularly when they do not share the same utility function. Because of the difficulty in performing interpersonal comparisons of utility, economists customarily think of utility as an *ordinal magnitude*, meaning that the absolute magnitude of utility is meaningless, but the relative magnitudes of utilities at various rates for an individual session define preferences among rates,[2] and the relative differences in magnitude indicate the strength of the preferences [4]. A consequence of considering only ordinal magnitudes is that utility functions are unique only up to a linear transformation. That is, the utility maximizing behavior of an individual with utility function $u(x)$ is indistinguishable from that of one whose utility function is a linear transformation of $u(x)$. This restriction makes intuitive sense because a linear transformation simply represents a change in scale and a translation of the zero point of the utility function, leaving the underlying preference relation unaffected.

The notion of an aggregate utility function is a compelling extension of the concept of individual utility. Aggregate utility is defined by a *social welfare function* (SWF) that maps the vector of all session utilities to a scalar value representing the social desirability of the corresponding vector of rates. The ranking of all possible rate allocations according to their aggregate utility is known as the *social preference relation* (SPR). As with individual utility functions, we are primarily interested in this preference relation rather than the absolute magnitude of the SWF.

In optimization-based congestion control, we adopt, somewhat arbitrarily, the sum of individual utilities as the SWF. In general, there are many ways to define the SWF, each carrying with it some subjective judgment about how individual preferences should be combined to determine a social preference.

Challenges for Multicast Extension

Naive Extension Is Flawed

The optimization-based approach described above has formed the basis for a flurry of research resulting in novel decentralized frameworks for unicast congestion control [3, 5, 6]. One important outcome of this research has been a deeper understanding of fairness, specifically the relationship between particular utility functions and formal definitions of fairness [5]. Given the importance of fairness in multicast congestion control, it is natural to wonder if these ideas could easily be extended to multicast. One reason to believe this is that the topologies of sessions are not explicitly modeled in the optimization-based framework. The links used by a unicast session presumably form an acyclic path between a source and a receiver. However, a session can use any subset of network links — for example, a tree in the case of multicast. In fact, single-rate multicast sessions are indistinguishable from unicast sessions within the mathematical framework. Thus, the solution of the congestion optimization problem is unaffected by their presence. In light of these observations, it appears that the optimization-based framework is readily generalized to single-rate multicast.

While it is true that, mathematically, single-rate multicast presents no new challenges for this theory, an uncritical application of the results in the multicast case leads to a serious problem: the resulting rate allocations are extremely unfair to multicast. The reason for this unfairness is that multicast is penalized for resource use without

[2] In this section we assume that utility is a function of session rate; we do so for the sake of concreteness and continuity with the rest of the article. It should be understood, however, that the current discussion applies to any utility function.

being rewarded for sharing bandwidth. In the naive extension, multicast and unicast sessions have identical utility functions. However, a large multicast session typically uses many more links than would a unicast session between the source and any single receiver; it can therefore expect to see a higher session price than the unicast session. Since session rate is a decreasing function of price, multicast sessions will consequently receive lower rates than unicast sessions along the same end-to-end paths.

We are thus faced with an incentive problem and a fairness problem.

Multicast Incentive Problem. Every multicast receiver has an incentive to leave the multicast session and establish a private unicast session with the sender, since doing so would yield a higher rate. However, it is in the network's interest to encourage the use of multicast because a multicast session demands far fewer network resources than a set of unicast sessions to the same receivers. A stable multicast service is unlikely to thrive when the interests of the network and its users are in opposition. For multicast to be widely used, users must have an incentive to choose multicast over unicast.

Unicast Fairness Problem. To solve the incentive problem, multicast receivers should expect a rate as high as or higher than they would obtain with unicast. However, such an incentive must be provided in a way that is fair (or at least not too unfair) to unicast traffic.

Below we describe an approach to the multicast incentive problem that has two components. First, we modify the definition of a session utility function to give more weight to sessions that serve more receivers. Second, we partition multicast sessions into smaller sessions, separating highly constrained receivers from less constrained ones. We then turn to the unicast fairness problem. Our solution is to define a mechanism for *controlled unfairness* to unicast, with a limiting case of strict fairness between multicast and unicast.

Redefining Multicast Utility

At the heart of the unfairness to multicast discussed earlier lies a subtle problem concerning the definition of the utility function for an individual multicast session. For a unicast session, it makes little difference whether we associate utility with the sender or receiver. For the purpose of unicast congestion control, we can treat the sender's and receiver's objectives as being one and the same. For a multicast session with multiple receivers it is unclear whether session utility belongs to the sender or is split in some way among the receivers.

Conceptually, our solution is to modify the social welfare function. We observed earlier that, while the sum of utilities is the SWF chosen to represent the aggregate utility, other SWF definitions are possible. In principle, we should be open to any alternative definitions for the aggregate utility of a session. To obtain tractable formulas, we consider a restricted set of definitions by attributing an identical utility function to each receiver[3] and combining these to get a weighted utility function for the session. Under this restriction, the SWF in the congestions optimization problem can be any *weighted* sum of utilities

$$\text{aggregate utility} = \sum_{\text{all sessions } i} f(R_i)u(x_i),$$

where the *weight function* $f(R_i)$ depends on the number of receivers R_i in session i. The optimal set of session rates depends on the specific weight function used to define this weighted sum. The original congestion optimization problem is obtained by assigning a constant weight to each session utility. We refer to a constant-weighted utility function as a *sender-oriented* definition of utility since one can think of it as associating all of the session utility with the sender regardless of group size. A fairly natural alternative is to make session weights directly proportional to the number of receivers $f(R) = \text{constant} * R$, which we refer to as a *receiver-oriented* definition. Although we could clearly consider many other dependencies of weights on group size, it turns out that the receiver-oriented definition has a very desirable property in the context of the session splitting problem, as we will now see.

For the purpose of unicast congestion control, we can treat the sender's and receiver's objectives as being one and the same. For a multicast session with multiple receivers it is unclear whether session utility belongs to the sender or is split in some way among the receivers.

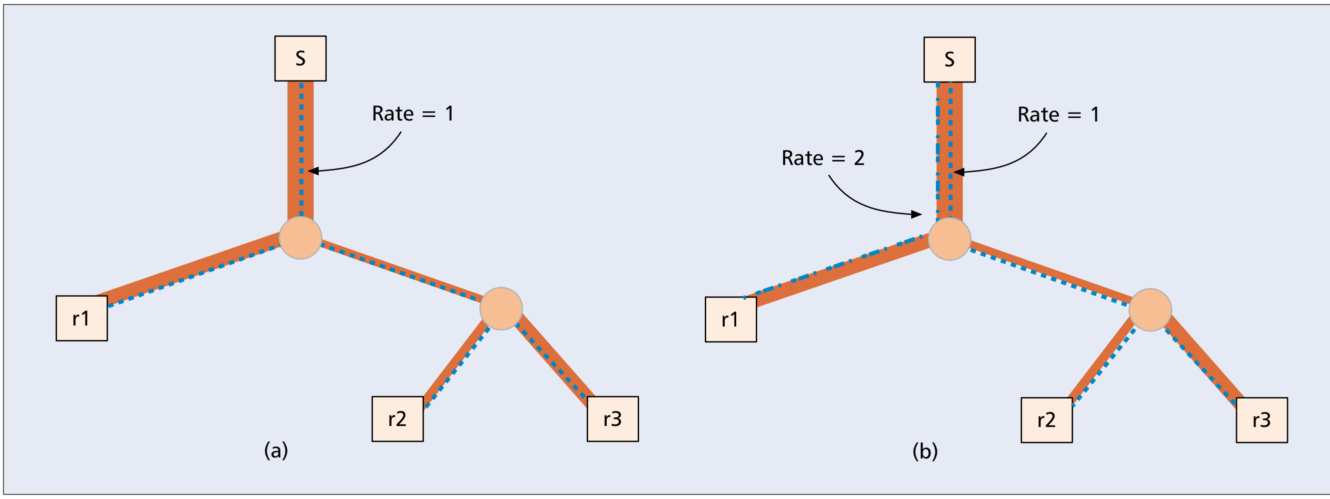
Session Splitting

We have asserted above that the most bandwidth-constrained receiver in a group should receive a rate at least as large as a unicast session from the source to the same receiver. However, even when the most constrained receiver is treated fairly in this way, other less constrained receivers still may get higher rates by defecting from the group and establishing a unicast session with the source. When the bottleneck bandwidth varies greatly for different receivers, low-bandwidth receivers can seriously constrain high-bandwidth receivers in the same session. In such cases, it is advantageous to split the session into two or more smaller sessions, each of which contains comparably constrained receivers, as long as the additional traffic introduced does not further congest the network.

Let us then focus on the abstract problem of splitting a session, namely, given a set of receivers, to determine the best way to partition them into sessions. Intuitively, we do not expect the limiting case of assigning each receiver to a separate unicast session to be satisfactory due to the congestion that would likely occur. As we have seen, however, a single large session may not be the best solution either. If we consider the multicast session in Fig. 2a, for example, it is easy to see that splitting the session as shown in Fig. 2b is desirable since it allows receiver r1 to obtain a higher rate without affecting the rate for receivers r2 and r3. In the general case, choosing the best partitioning of receivers is nontrivial, and we require some principled way of selecting the best out of all possible ones.

A session-splitting approach follows naturally from adopting the same utility maximization approach used for congestion control. For this problem we consider a modified network model in which all sessions other than the one to be split are removed, and the link capacities are interpreted as the capacities available for that particular session. Thus, the effect of splitting on unicast sessions is not considered. The optimal partition of the receiver set is the one that maximizes the aggregate utility of all sessions. It is clear that the optimal

[3] *To the extent receivers within a session share the same application requirements, it is also reasonable to assume they share a common utility function.*



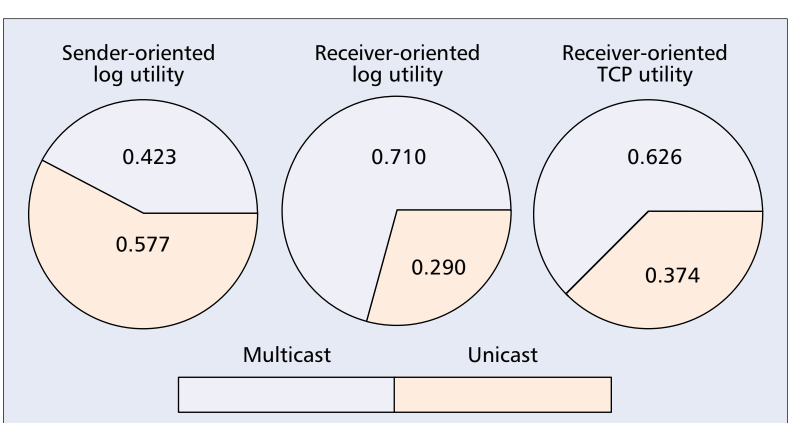
■ **Figure 2.** *An example of session splitting. a) a single session b) becomes two sessions.*

partition will depend on the social welfare function we choose, since this definition determines the weights of each session's utility function.

In keeping with the standard interpretation of utility functions in economics, we pose as an axiom that the optimal partition should be invariant under a linear transformation of utility. It seems rational that since the preferences of individual sessions are invariant under such a transformation, that the same invariance should hold for the social preference relation, which we defined in an earlier section. Thus, we claim that this axiom should be satisfied under any rational definition of session utility. It turns out that *only* receiver-oriented utility functions satisfy this axiom. Thus, receiver-oriented utility functions are, in this particular sense, the only rational definition of utility for single-rate multicast.

CONSEQUENCES OF RECEIVER-ORIENTED UTILITY

In the previous two sections, we refined the congestion optimization problem by introducing receiver-oriented utility functions, which we justified on the basis of an invariance property in the context of the session splitting problem. We now return to the congestion control problem and consider the impact of receiver-oriented utility on fairness. We want to ensure first that the problem of unfairness to multicast is addressed by this change, and second that the resulting rate allocations are not excessively unfair to unicast.



■ **Figure 3.** *Optimal rate allocations at the bottleneck link for the example of Fig. 1b for three different definitions of session utility.*

By solving the optimality conditions for the modified congestion optimization problem (where we maximize a weighted sum of utilities), we find that in the optimal rate allocation, the rate of each session is a decreasing function of price (as before) but an increasing function of group size. In order to understand the resulting fairness to small groups (with unicast as the limiting case of a single receiver) it is necessary to understand the relative strengths of these two dependencies (on price and group size).

Figure 3 compares the bandwidth allocated to the multicast and unicast sessions at the bottleneck link of Fig. 1b under three definitions of session utility.

The left pie chart illustrates the lack of incentive for receiver 3 to use multicast under sender-oriented utility. We see in the middle and right pie charts that under receiver-oriented utility the multicast session gets a larger share of bandwidth on the bottleneck link than the competing unicast session. The resulting allocations address the multicast incentive problem, but become extremely unfair to unicast.

CONTROLLING UNICAST UNFAIRNESS

We take the position that some amount of unfairness to unicast is desirable (and, in a sense, inherent) since it provides an incentive to use multicast. A receiver in a multicast group can expect to obtain a higher rate than it would by leaving the group and establishing a unicast session. At the same time, it is essential to ensure that unicast traffic will not starve if multicast is to be deployed on real networks.

We start by making the following observation, which we illustrate in the context of our sample topology. The degree of unfairness to unicast is dependent on the form of utility function we attribute to sessions. Notice in Fig. 3, for example, that the multicast rate is lower under the

TCP utility function than under the logarithmic utility function, although both are unfair to the unicast session. This observation, which holds in more general topologies, suggests that we can control the degree of unfairness to unicast by selecting an appropriate utility function while retaining the receiver-oriented definition of session utility.

Consider the family of utility functions defined by a single parameter α:

$$u(x;\alpha) = \begin{cases} \log x & \alpha = 0 \\ -\dfrac{x^{-\alpha}}{\alpha} & \alpha > 0 \end{cases}.$$

Notice that this family includes both logarithmic utility (α = 0) and TCP utility (α = 1). The parameter α provides a mechanism to tune the unfairness of the optimal rate allocations. It turns out that as α increases, the resulting rate allocation approaches a form of fairness known as *max-min fairness*.

The max-min fair rate allocation is a unique feasible allocation having the property that the rate of any session cannot be feasibly increased without decreasing the rate of another session whose rate is already lower. It can be shown that under max-min fair allocation, a multicast session and unicast session sharing the same bottleneck will receive equal rates. We may think of max-min fairness as being strictly fair to both unicast and multicast.

Thus, parameter α provides a "control knob" we can use to tune the unfairness to unicast, while retaining the incentive to use multicast. Figure 4 illustrates the effect of changing α in our running example. The plot shows the ratio of rates for multicast and unicast sessions sharing the bottleneck link. This ratio is equal to one in the case of max-min fairness and approaches one asymptotically as α increases. This figure suggests that a relatively small value of α may be sufficient in practice since the ratio approaches one very quickly at low values of α but flattens for larger values.

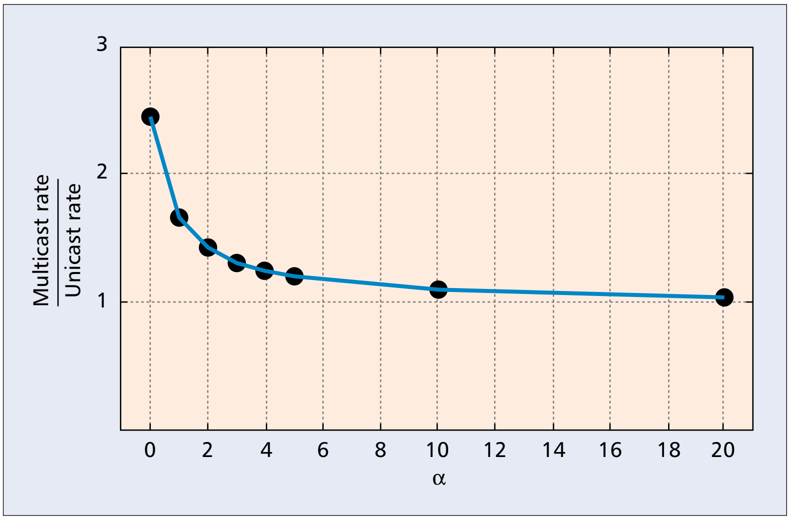


Figure 4. *Ratio of multicast to unicast rate on the bottleneck link of Fig. 1b as a function of parameter α. As α increases, the two sessions approach equal sharing of the link, but multicast is always favored.*

Conclusion

This article presented the application of optimization-based congestion control to multicast, enabling a principled approach to the important problem of fairness between multicast and unicast traffic. We identified a fundamental trade-off between the performance expectations of users and the stability concerns of network operators. To provide an incentive to use multicast, we introduced the receiver-oriented definition of session utility, which we argued is the only definition to satisfy an important invariance property in the context of session splitting. However, adopting the receiver-oriented definition resulted in unfairness to unicast sessions. We then introduced a parameterized utility function that allows protocol designers to select an acceptable level of unfairness, while still providing the necessary multicast incentive. Much work still remains in the design of practical protocols based on this foundation as the quest continues for Internet-wide multicast deployment. One exciting area for future research is how to exploit existing packet marking mechanisms for communicating prices along a simple path to compute the session price for a multicast tree.

References

[1] C. Diot *et al.*, "Deployment Issues for the IP Multicast Service and Architecture," *IEEE Net.*, vol. 14, no. 1, Jan./Feb.2000, pp. 78–88.
[2] K. L. Calvert *et al.*, "Concast: Design and Implementation of and Active Network Service," *IEEE JSAC*, vol. 19, no. 3, 2001.
[3] S. Kunniyur and R. Srikant, "End-to-end Congestion Control: Utility Functions, Random Losses and ECN Marks," *Proc. INFOCOM*, 2000.
[4] J. Hirshleifer and D. Hirshleifer, *Price Theory and Applications*, 6th ed., Prentice Hall, 1997.
[5] F. P. Kelly, "Charging and Rate Control for Elastic Traffic," *Euro. Trans. Telecommun.*, vol. 8, 1997 pp. 33–37.
[6] S. H. Low and D. E. Lapsley, "Optimization Flow Control, I: Basic Algorithm and Convergence," *IEEE/ACM Trans. Net.*, Dec. 1999.

Biographies

JONATHAN SHAPIRO (jshapiro@cs.umass.edu) graduated from Columbia University (1990) with a B.A. in physics. He worked for eight years as a software developer before returning to school at the University of Massachusetts at Amherst. He was awarded an M.S. in computer science in 2000 and is currently a Ph.D. candidate. His research interests include congestion control and quality of service, and the role of incentives in distributed computing.

DON TOWSLEY (towsley@cs.umass.edu) received his B.A. degree in physics and Ph.D. degree in computer science from the University of Texas in 1971 and 1975, respectively. From 1976 to 1985 he was a member of faculty of the Department of Electrical and Computer Engineering at the University of Massachusetts, Amherst. He is currently a professor of computer science and co-director of the Computer Networks Laboratory at the University of Massachusetts. His research interests include high-speed networks, multimedia systems, and stochastic scheduling.

JIM KUROSE (kurose@cs.umass.edu) received a B.A. degree in physics from Wesleyan University in 1978, and M.S. and Ph.D. degrees in computer science from Columbia University in 1980 and 1984, respectively. He is currently a professor (and past chairman) in the Department of Computer Science at the University of Massachusetts, where he is also co-director of the Networking Research Laboratory. His research interests include multimedia communication, network protocols, network and operating system support for servers, and modeling and performance evaluation.