

Applying the Generalized Vickrey Auction to Pricing Reliable Multicasts

Ashish Sureka¹ and Peter R. Wurman¹

North Carolina State University, Raleigh, NC 27695, USA,
asureka@unity.ncsu.edu, wurman@ncsu.edu

Abstract. Reliable multicast networks scale back the speed at which content is delivered to accommodate the slowest receiver. By extracting information about the receiver's capabilities, the service provider can select which receivers to admit, and at which speed to operate the network. We examine the application of the Generalized Vickrey Auction to pricing reliable multicast and present a distributed method to compute the GVA prices.

1 Introduction

Multicast protocols allow a message to be sent to many receivers at the same time more efficiently than a unicast protocol and less wastefully than broadcast protocols. In a multicast network, the messages are sent only to identified subscribers. The rate at which any individual subscriber receives data depends on many factors, including the type of machine and the throughput of its connection to the internet. In standard multicast protocols, hosts or network elements that cannot handle data at the rate being sent by the server will simply drop packets.

While packet dropping is satisfactory for some applications, in some situations the sender wants to be sure that everyone receives all of the data in a timely manner. Potential applications that require this reliability include some uses of video conferencing, stock quote feeds, auctions, online computer games, and other real-time applications that demand symmetric and complete information delivery. *Reliable multicast* protocols are designed to satisfy this need. In order to achieve reliability and source-order data delivery, the sender must send at a speed acceptable to the slowest receiver. Thus, the rate at which all subscribers receive packets is determined by the capabilities of the slowest subscriber.

We assume that the service provider has the ability to charge subscribers for the content, either to recover her costs (and make a profit) or to discourage network abuse. In addition, each subscriber has a value for the service that depends upon the speed at which it is offered. The service provider must determine both the speed and the price at which to offer her service. From the point of view of the provider, the marginal cost of adding subscribers is zero. Thus, we assume that the service provider will choose to admit anyone into the subscriber pool who has the capability to receive data at a rate at least as fast as the provider's chosen speed. However, the provider runs the risk of having a user overstate his reception rates in order to gain admittance to the subscriber pool. Once the user is in the pool, the reliable multicast protocol will automatically

compensate for his slower speed and failure to acknowledge packets by decreasing the sending rate for everyone.

In this paper, we consider the task of admitting potential receivers into the subscriber pool and selecting a service speed that maximizes the social welfare (as opposed to maximizing the revenue). The task is simplified greatly if the service provider can extract the true capabilities and values from potential subscribers. The Generalized Vickrey Auction (GVA), a mechanism well-known in the economics literature, has the property that it is a participant's weakly dominant strategy¹ to truthfully reveal his value for, in this case, receiving data at various speeds and the limitations of his capabilities.

Although in general the GVA is NP-hard, we demonstrate that in the case where there are a linear number of potential operating speeds, it is computationally feasible for the agents to report their valuations for the different network speeds and for the service provider to determine the various payments. In addition, we show that the computation can be distributed across the network in a manner that reduces the overall number and size of the messages that must be communicated.

2 Model

We consider a simple model of a reliable multicast network. The network is a tree rooted at the service provider. The leaves of the tree represent clients who have values for receiving the multicast. The internal nodes of the tree represent the routers.

In the general model, the service provider, k , is capable of transmitting the signal at a speed chosen from the set S . Let I represent the set of possible receivers. An individual receiver, i , has a value for receiving the transmission at speed s denoted $v_i(s)$. Let J be the set of routers, with an individual router denoted j .

Receiver i has a threshold, t_i , above which he cannot keep up with the transmission. If the service is provided at a speed greater than t_i , receiver i will start dropping packets and the reliable multicast protocol will automatically rein back its delivery speed. Thus, the service provider would like to know the value of t_i for each potential recipient. It is convenient to model the threshold as $v_i(s) = 0$ for all $s > t_i$.

The service provider has a cost for broadcasting at speed s denoted $\gamma(s)$. We consider the case where the service provider desires to offer the service at the speed that maximizes the social welfare. That is,

$$V^* = \max_s -\gamma(s) + \sum_{i \in I} V_i(s). \quad (1)$$

Let s^* be the speed resulting in V^* . Every receiver capable of receiving data at or above speed s^* is admitted.

¹ In a game a strategy is said to weakly dominate all other strategies in the player's strategy set if, regardless of what he expects his opponents to do, this strategy always yields a payoff that is at least as good as any other of his strategies.

2.1 Generalized Vickrey Auction

The Generalized Vickrey Auction (GVA) [9] is a well-known mechanism in economics which has the surprising and desirable property that each participant has a weakly dominant strategy to truthfully reveal his true values for all options. Informally, the GVA achieves this property, called *incentive compatibility*, by computing a payment for an individual participant that is independent of his stated values; the participant's statement determines whether she will receive anything, but not how much she will pay.

To compute the GVA payment, the service provider computes the allocation V_{-i}^* that maximizes the sum of utilities of the other participants when i is excluded from the allocation.

$$V_{-i}^* = \max_s -\gamma(s) + \sum_{h \in I, h \neq i} v_h(s).$$

Receiver i 's payment when speed s^* is chosen is

$$p_i = V_{-i}^* + \gamma(s^*) - \sum_{h \in I, h \neq i} v_h(s^*). \quad (2)$$

The GVA computes a payment per individual, not a price per resource. Thus, two individuals can pay different amounts for the same resource, a fact that is particularly clear in the reliable multicast setting in which all admitted receivers get the same data at the same rate.

In many practical settings, the service provider would be more concerned with selecting the speed and the pricing method that will generate the greatest profit (revenue minus cost). However, there are arguments in the economics literature [2] that suggest that socially efficient mechanisms will generate as much revenue as those that try to maximize revenue because efficient mechanisms will attract greater participation and competition.

Economic arguments aside, in this paper we are concerned with removing the incentive to misrepresent operating capabilities that will later degrade the experience for the entire user population. By using an incentive compatible mechanism, we can extract truthful information from the participants regarding their values and capabilities and set the operating speed appropriately.

Example 1 The simple example shown in Figure 1 has a service provider, K, seven receivers (R1 to R7) and 6 routers (N1 to N6). For simplicity, assume the service provider can transmit the data at three different speeds (`fast`, `medium`, and `slow`). Table 1 shows the costs for the service provider and the values for each of the receivers for each of the three speeds. The network on the right branch of the tree is slower than the left, and the receivers on the right cannot receive at the `fast` rate. In place of the threshold, we have set the right receivers' utility for the `fast` speed to zero.

It is easy to see that `fast` is the optimal speed and produces a total utility of 39 (versus 37 for `medium` and 11 for `slow`). The GVA payments are computed according to equation (2). For receiver R1, we first compute the value of the optimal solution that is credited to other agents: $V^* - v_{R1}(s^*) = 39 - 9 = 30$. Next, we compute the utility the other agents would have if R1 were not present. Without R1, the optimal speed is

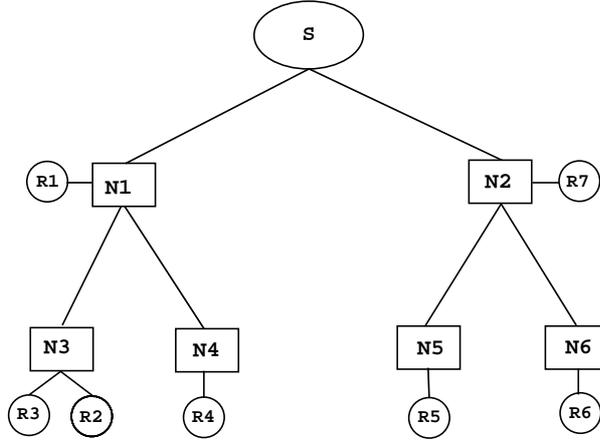


Fig. 1. Example network model.

	Fast	Medium	Slow		Fast	Medium	Slow
K	(3)	(2)	(1)	R4	11	3	1
R1	9	4	1	R5	0	8	1
R2	14	7	2	R6	0	7	2
R3	8	6	3	R7	0	4	2

Table 1. The costs for the service provider K, and the utilities of receivers R1 to R7 for three speeds.

medium and $V_{-i}^* = 33$. Thus, R1’s GVA payment is 3. Similarly, we can compute that R2 pays 5, R4 pays 6, and all of the other agents pay zero.

3 A Distributed Protocol for Computing Payments

A straightforward implementation of a GVA mechanism in this environment is for each receiver to send his utility function to the “auctioneer”, either the provider or a designated third party. The auctioneer then computes each receiver’s payment and informs each receiver individually. This scheme has several drawbacks. First, as we shall see below, the amount of information that needs to be communicated is unnecessarily high. Second, the details of each user’s utility function is passed all the way to the top of the tree. An observer near the top of the tree would see the utility function of every user on that branch. The alternative scheme described below aggregates information, thereby making it more difficult for an observer to discern a particular agent’s utility.

For the reliable multicast network model, we can compute the GVA payments in a distributed fashion. The solution computes each receiver’s payment in the router at which they are connected to the tree. For the router to make this computation, it must know the value of operating at all speeds, and the effect of adding this particular receiver on the choice of network speed. Let I_j be the set of receivers directly connected to

router j , $C(j)$ be the routers directly connected to j as child nodes, and $p(j)$ be the parent node of router j . Note that all of the receivers not connected to j can be reached by a traversal through one of the nodes in the set $C(j) \cup p(j)$. Let $I_{p(j)}$ and $I_{c(j)}$ be the set of receivers reachable from j through the parent and a child, c , respectively. We can rewrite equation (1), from the perspective of receiver j , as

$$V^* = \max_s -\gamma(s) + \sum_{i \in I_j} V_i(s) + \sum_{i \in I_{p(j)}} V_i(s) + \sum_{c \in C(j)} \sum_{i \in I_{c(j)}} V_i(s). \quad (3)$$

We can rewrite equation (2) in a similar fashion. It is clear from the above decomposition that to compute the GVA payment of $i \in I_j$, router j needs to know only the aggregate utility at each speed for all of the nodes reachable through its parent and its children. To make the following description more concise, let $V_{p(j)}$ be the vector of aggregate demands for the nodes reachable through j 's parent, $V_{c(j)}$ the aggregate demand for the nodes reachable through j 's child c , and V_{I_j} the aggregate demand of receivers directly connected to j . Our distributed protocol computes the necessary aggregate utility messages and propagates them through the network.

The protocol works as follows:

Step 1: Every receiver sends its utility vector to the router connecting it to the multicast tree. Each router j now has v_i for all $i \in I_j$ and can compute V_{I_j} .

Step 2: Each router with no children sends V_{I_j} to its parent. When a router with children receives $V_{c(j)}$ for each of its children, it computes $V = V_{I_j} + \sum_{c \in C(j)} V_{c(j)}$ and sends it to its parent. This step propagates aggregate information up the tree.

Step 3: Each router propagates aggregate information about the rest of the tree to each of its children. The composition of router j 's message to its child c depends upon router j 's position in the tree, as follows:

$$V = \begin{cases} V_{I_j} - \gamma + \sum_{z \in C(j), z \neq c} V_{z(j)} & \text{if } j \text{ is the root,} \\ V_{I_j} + V_{p(j)} + \sum_{z \in C(j), z \neq c} V_{z(j)} & \text{otherwise.} \end{cases}$$

This step propagates information both sideways and down the tree. The service provider may choose to include the optimal operating speed as part of the message, though every router will have enough information to compute it directly.

Step 4: At this point, each router has enough information to compute the payment for each receiver attached to it. The reader can verify that router j now has all of the terms required in equation (3). Once the router has computed the payments, a message of the form $\{i, p_i\}$ must be sent to both the receiver and the service provider.

The GVA computation is decomposable in this setting because the number of solutions is the number of operating speeds, and the "resource" is *non-rival*, that is, the object can be assigned to more than one participant. In fact, the very purpose of this enhancement to the reliable multicast system is to determine a common speed at which to operate. In the general case of assigning combinations of rival objects, all possible sub-solutions would have to be communicated throughout the tree, making the approach unworkable.

3.1 Example of the Distributed Protocol

To illustrate the protocol, consider router N1 in the example problem defined in Figure 1 and Table 1. During Step 1, N1 learns that $v_{R1} = [9, 4, 1]$. In Step 2, N1 receives message $[22, 13, 5]$ from N3 with the aggregate values of receivers R2 and R3, and message $[11, 3, 1]$ from N4. N1 then composes its message to the service provider, K, by summing the vectors it has received, producing $[42, 20, 7]$. The service provider receives $[42, 20, 7]$ from N1 and, after the protocol on the right side of the tree completes, $[0, 19, 5]$ from N2.

In Step 3 of the protocol, K constructs the message $[-3, 17, 4]$ to send to N1 by subtracting the cost vector from the message from N2. N1 now has the information shown in Table 2, which is exactly what it needs to compute that R1's GVA payment is 3.

	Fast	Medium	Slow
R1	9	4	1
N3	22	13	5
N4	11	3	1
K	-3	17	4

Table 2. Information at router N1 after Step 3 of the protocol.

Router N1 was also responsible for passing information down the tree in Step 3. N1's message, $[17, 24, 6]$, to N3 is the sum of the vectors describing R1, N4, and K. Similarly, N1 sends $[28, 34, 10]$ to N4. These two messages give N3 and N4, respectively, enough information to compute the GVA payments for R2, R3, and R4.

3.2 Computational Considerations

Above, we claimed that the distributed protocol had some advantages over the straightforward implementation in which all utility vectors are passed up the tree to the service provider, who then computes payments and sends them back down the tree to the receivers. In this section we elaborate on those claims.

Consider a symmetric tree of depth d in which each router has b children and l directly connected receivers. We will distinguish the computation phase, which ends when some element of the network knows the payment for each receiver, and the notification phase, which ends when both the service provider and receiver know the payments.

The communication costs can be measured in both the amount of information communicated, and the amount of time taken to perform the communication. First, we note that the notification phase has the same cost in both the centralized and distributed protocols because a complete message path has to exist between the service provider and the receiver to carry the payment notification to both parties. It does not matter whether the message starts at the service provider or the router near the receiver.

Thus, we concentrate on the amount of information communicated in the computation phase of the two protocols. We can eliminate the communication between the

receivers and their local routers, as this is the same in both protocols. However, with respect to information passed among the routers, the distributed protocol has significant savings over the centralized protocol. In the straightforward implementation, the number of vectors that a router in level n of the tree must send to its parent is $l\left(\frac{b^{d-n+1}-1}{b-1}\right)$. In contrast, the distributed protocol sends two vectors (one each in Steps 2 and 3).

Although the distributed protocol requires far fewer vectors to be communicated, it does require more traversals of the tree. The centralized protocol requires two traversals, one up from the bottom with all of the utility functions, and then one down from the top with all the payments. The distributed protocol requires three traversals, one each in Steps 2, 3, and 4. It seems likely that the benefits of reducing the number of messages that need to be communicated will compensate for the increased number of tree traversals.

In the centralized protocol, no information needs to be stored in the routers; each message can be sent directly up the tree and forgotten. In contrast, in the distributed protocol, every router needs to store information about the reported utilities in order to later determine the payments for the receivers connected to it. However, the router needs to store a utility, or aggregate utility, vector from only those receivers and routers directly connected to it. In the symmetric case, the router must store $b + 1 + l$ vectors of size $|S|$.

3.3 Other Considerations

One of the desirable characteristics of a pricing mechanism is its robustness to changes in the subscriber population. If the payment of one receiver changes drastically because another receiver left or joined the network, then the pricing mechanism is not robust. With GVA pricing, when a new receiver joins or leaves the network, the service provider recalculates the optimal speed at which the network should operate and determines the payments for each of the receivers. The addition or subtraction of one receiver can change the payments of other receivers and even cause the service provider to select a new speed, abandoning some of the previous recipients. The former problem is not serious, as the new payments will never be greater than a receiver's value for the service. However, the latter issue is of some concern. While we have not yet studied this problem, we suspect that a utility model that properly considers both the value of admission and the value of continuing service over time is a promising avenue to explore.

Two well-known drawbacks of the GVA are its susceptibility to *collusion* and *false-name bids*. Bidder collusion is a serious problem in many auctions; in our proposed mechanism the bidders can collude by deciding beforehand on values to report to the auctioneer that improve the utility of at least one of the agents. Table 3 shows the valuations of three receivers for three speeds. If the receivers truthfully report this information, the respective payments will be 4, 0, and 0. No receiver can unilaterally change the vector he reports and make himself better off.

However, if Receivers 1 and 3 collude and report the values shown in Table 4, the respective payments will be 3, 0, and 0. Thus, by reducing his stated value for medium and slow, Receiver 3 is able to lower Receiver 1's payment. Presumably, Receiver 1 will be willing to offer Receiver 3 some portion of the savings as a reward, thus improving both agents' overall utility. In the same spirit, it is possible for a receiver to

	Fast	Medium	Slow
Receiver 1	9	1	1
Receiver 2	0	6	3
Receiver 3	4	2	2

Table 3. Information at the service provider.

	Fast	Medium	Slow
Receiver 1	9	1	1
Receiver 2	0	6	3
Receiver 3	4	1	1

Table 4. Information at the service provider.

submit a bid under a false name and lower her payment [10]. However, to succeed with this charade in the reliable multicast scenario would also involve running, or spoofing, a real multicast client or the false bid would be quickly detected.

One final potential drawback of calculating payments using the GVA is *price discrimination*. Two agents might end up paying different prices for the same speed even when they have the same value for receiving data at that speed. There are likely to be reliable multicast applications in which price discrimination will be regarded as unfair and unacceptable by the receivers. The perceived negative benefits of price discrimination will have to be weighed against the potential loss of social welfare due to the loss of incentive compatibility.

4 Related work on network pricing schemes

Several researchers have proposed new ways of controlling network usage using economic mechanisms. The approaches address the questions of how to allocate bandwidth in a network and how to do admission and congestion control.

Smart Markets were proposed by MacKie-Mason and Varian [5] and introduced the concept of congestion-sensitive pricing. Rather than forwarding packets based on a FIFO policy, packets are selected to be forwarded based on the priority (or bids) associated with each packet. The proposed Smart Market is a GVA, and achieves incentive compatibility by charging the successful senders the priority of the highest priority packet that is dropped. However, because it requires accounting at the level of packets, the Smart Market is impractical to implement in a real network [8].

A experimental evaluation was done by Breker [3] on both static and dynamic pricing strategy. Breker's simulations show that when strategies are static, as the price per packet increases, the network utilization decreases. In the dynamic pricing strategy simulations, the price varies over time as the result of user demand for bandwidth. Breker also found, predictably, that as the number of users increases, the price per unit of bandwidth also increases.

Murphy et al. [6] propose a feedback scheme to increase network efficiency by taking advantage of the fact that users can modify their traffic inputs in response to feedback signals from the network such as cell loss and/or delay. Each user computes a demand curve that specifies her willingness to pay for various levels of bandwidth. The service provider dynamically adjusts the prices based on the current network conditions and sets the prices so that the marginal benefit the user places on his resource allocation is equal to the marginal cost of handling the resulting traffic in the network.

Shenkar et al. [7] critiques the optimality paradigm (achieving optimal welfare requires charging marginal congestion costs for usage) and propose that the pricing in

computer networks should focus more on structural and architectural issues rather than the optimality paradigm.

Feigenbaum et al. [4] investigate cost sharing protocols for multicast transmission [4] and point to two distinct mechanisms, *Marginal Cost* and *Shapley value*, as the two solutions appropriate in the context of sharing the cost of a multicast transmission. In their paper, Feigenbaum et al. prove that *Marginal Cost* has a natural protocol that uses only two messages per link of the multicast tree and the Shapley Value requires a quadratic total number of messages. The paper analyzes the communication complexity of the two mechanisms and the apparent gap between the communication complexity of the Marginal Cost and Shapley Value.

Our protocol also bears some resemblance to the COTREE algorithm proposed by Andersson and Ygge [1] which propagates aggregate (continuous) demand messages through a self-imposed communication tree. The COTREE algorithm is designed to propagate the necessary aggregate demand information to the root of the tree so that an equilibrium price can be computed.

5 Conclusions

In this paper we applied the Generalized Vickrey Auction to the problem of allocating bandwidth in a multicast network, and argue that it will lead to an efficient use of the resources. The computation of GVA payments in this environment can be efficiently computed in a distributed fashion that greatly reduces the amount of information that must be propagated through the network.

In the future, we plan to examine pricing in a more detailed model of reliable multicasting, and to investigate robustness of the mechanism against collusion and false name bids.

Acknowledgements

This work was supported by the Defense Advanced Research Projects Agency under contract F30602-99-1-0540 and E-Commerce@NC State program. The views and conclusions contained herein are those of the authors. We would like to thank Doug Reeves for his valuable comments.

References

1. Arne Andersson and Fredrik Ygge. Managing large scale computational markets. In *31st International Conference on System Sciences*, pages 4–13, 1998.
2. Lawrence M. Ausubel and Peter Cramton. The optimality of being efficient. May 1998.
3. Leanne P. Breker. A survey of network pricing schemes. Technical report, Department of Computer Science, University of Saskatchewan, 1996.
4. Joan Feigenbaum, Christos H. Papadimitriou, and Scott Shenker. Sharing the cost of multicast transmissions (preliminary version). In *ACM Symposium on Theory of Computing*, pages 218–227, 2000.

5. J. MacKie-Mason and H. Varian. Pricing the internet. Technical report, In Public Access to the Internet (Kahn B. and Keller J., Eds.), Prentice Hall 1995., 1995.
6. L. Murphy and J. Murphy. Pricing for atm network efficiency. Technical report, Proc. 3rd International Conference on Telecommunication Systems Modelling and Analysis, Nashville, TN, March, p. 349-356. Available from URL <http://www.eeng.dcu.ie/murphyj/atm-price/atm-price.html>, 1995.
7. S. Shenker, D. Clark, D. Estrin, and S. Herzog. Pricing in computer networks: Reshaping the research agenda. Technical report, Proc. of TPRC 1995., 1995.
8. R. Singh, M. Yuksel, S. Kalyanaraman, and T. Ravichandran. A comparative evaluation of internet pricing models: Smart markets and dynamic capacity contracting. Technical report, Proceedings of 10th Annual Workshop on Information Technologies and Systems (WITS), Australia, 2000.
9. H. Varian and J. MacKie-Mason. Generalized vickrey auctions. Technical report, Technical report, University of Michigan., 1995.
10. M. Yokoo, Y. Sakurai, and S. Matsubara. Robust combinatorial auction protocol against false-name bids. *Artificial Intelligence*, 130(2):167–181, 2001.