A Sender-Adaptive & Receiver-Driven Layered Multicast Scheme for Video Over Internet

Quji Guo†, Qian Zhang, Wenwu Zhu, Ya-Qin Zhang
Microsoft Research China
5F, Beijing Sigma Center, No. 49, Zhichun Road
Haidian District, Beijing 100080, P. R. China

ABSTRACT

A new architecture of sender-adaptive and receiver-driven layered multicasting for video over Internet is presented in this paper. This architecture optimizes the overall performance by making the sender collaborate with the receivers. The statistical characteristics of bandwidth and packet loss ratio of all the receivers, the network temporal variation of each receiver, and the rate-distortion relation of the video codec have been taken into account. Techniques of network resource allocation, scalable video coding, FEC and pseudo-ARQ are also integrated in this scheme. Moreover, we introduce a new mathematical framework named as Q-Space Model & Optimization to generalize the multicasting approach. Simulation results demonstrate effectiveness of our scheme.

I. INTRODUCTION

Multicast is a promising technique for saving resource when delivering quality-adaptive multimedia over Internet with good scalability. One of the main challenges multicast leaves to application designers is how to support heterogeneous receivers. Excessively high bit rates would result in congestions to the users with lower bandwidth. While conservatively low bit rates would waste most users' bandwidth. In order to solve the rate control problem in multicasting, there exist two types of approach to date: sender-driven and receiver-driven.

The typical receiver-driven approach is the Receiver-driven Layered Multicast proposed by McCanne et al. [1]. The key idea is to generate several multicast groups for layered video source such that the receivers can get video data with different quality by subscribing different groups according to their network conditions and subscribing criteria. However, if a receiver attempts to join a group without enough bandwidth, congestion would occur and other receivers sharing the same bottleneck link will be affected. Some improvements were proposed in [2], though explicit estimation of available bandwidth was not considered.

In sender-driven approaches, sender adjusts data rates according to feedbacks [3-5]. Only one layer was considered in [3]. In [4] Vickers et al. introduced a Source Adaptive Multi-layered Multicast algorithm where the source uses congestion feedback to adjust the number of layers and the bit rate of each layer. Feedback mergers are used throughout the network. Cheung et al. [5] proposed Destination Set Grouping algorithm, which shares both receiver-driven and sender-driven approaches. However, by using independent streams rather than layered streams, it leads to an inefficient usage of the bandwidth. Note that all the above works did not take heterogeneous packet loss ratios into account.

For a good multicast scheme, several issues should be addressed. First, fairness is very important for Internet since a lot of heterogeneous traffics share the same communication resource. Padhye et al. addressed estimation formula for TCP throughput [6]. Rubenstein et al. addressed the impact of multicast layering on network fairness [7]. They identified some desirable fairness properties for multicast networks and pointed out that sender-coordinated layered protocols show promise for achieving desirable fairness. Secondly, packet loss protection is another important issue in multicast. Chou et al. addresses the possibility to use FEC and pseudo-ARQ for layered multicast within the receiver-driven framework [8]. However, in this open-loop system, parameter optimization is still hardly applicable due to lack of receiver information.

In this paper, we introduce a new architecture named Sender-Adaptive & Receiver-Driven Layered Multicast. It uses the approach as in the layered multicast with FEC and pseudo-ARQ to generate the multiple multicast groups. All receivers send sparse feedbacks containing statistical information of their network conditions to the sender for optimizing transmission parameters based on the Rate-Distortion relationship of each layer. On the other hand, the right of making subscribing decisions is still left to the receiver.

The rest of the paper is organized as follows: Section II gives a detailed description of the overall architecture. Section III analyzes and solves the optimization task on the sender side, which is the main focus of this work. Section IV discusses the receivers’ behaviors. Some simulation results are given in Section V. Conclusion and future work are presented in Section VI.

II. A NEW SYSTEM ARCHITECTURE FOR LAYERED MULTICASTING

Figure 1. An architecture for our sender-adaptive and receiver-driven layered multicasting.

Figure 1 depicts an architecture for sender-adaptive and receiver-driven layered multicasting. The sender encodes the video sequence into multiple data streams by using scalable source codec with FEC and then sends each stream as a separate multicast group. Different elementary stream, i.e., each multicast group, includes either source data in different layer (named as primary stream) and some parity check data (named as protection stream). Subscribing different combination of these multicasting groups means to get video data with different quality and different packet loss protection. The receivers subscribe or quit one or more groups from time to time according to their time varying network capabilities and packet loss ratios. Thus, this strategy is called as ‘receiver driven’. All receivers send sparse feedbacks, which are feedbacks...
containing statistical information about the receivers’ network conditions (e.g., available bandwidth, bandwidth variation, packet loss ratio, etc.) to a feedback analyzer. The feedback analyzer analyses all the feedbacks and find optimal parameters including data rate in each primary stream, channel coding parameters, etc. for the sender considering all the receivers and the Rate-Distortion relations. Then sender adjusts its parameters correspondently. Hence, this strategy is termed as ‘sender adaptive’.

Generally speaking, the sender’s role is to generate multiple streams according to the parameters given by the feedback analyzer, and then put each stream to a multicast group. The feedback analyzer is the critical component, which determines all parameters. The receiver monitors its network condition, reports it to the analyzer and most important, makes subscribing decisions.

In this scheme, the layered scalable source codec plays an important role so that the bit rate in each multicast group can be adjusted continuously. For each primary stream, the channel-coding module generates corresponding protection data and then split it into several protection streams, leading to different multicast groups. Figure 2 shows an example of how to generate elementary streams.

A receiver subscribing the source data also subscribes some of the protection streams in order to protect the packet loss. We expect that a receiver keep on subscribing some protection streams, which are necessary for most time, and subscribes some additional ones only when it experiences a higher instant packet loss ratio. To achieve this, some protection streams must be sent out later so that the receiver can have time to join a new group to get the protection data. This purposed delay can be different among protection streams and may be changed by the sender over time as well. In this paper we will not discuss how to choose delay parameters in great details. In the meanwhile, a receiver would try to access more source data by joining more multicast groups to get a higher quality when the bottleneck bandwidth becomes higher, and vice versa.

We do not require specific way for the analyzer to get feedback. A possible way to get feedback without any changes in current IP protocol at network level is to use UDP unicast. To avoid feedback implosion, each receiver must adjust the sending rate of feedback according to the control information given by the sender based on the total feedback density. However, if there are some special supports in networks, e.g., the feedback merger, people may implement novel methods to get feedbacks more efficiently without affecting the whole system architecture. Note that what we proposed here is an application level approach in order to avoid implosion.

In the following, we will mainly focus on how to generate multicast groups for a video sequence and how to determine the parameters for each group.

III. MULTICASTING-GROUP GENERATION AND PARAMETERS OPTIMIZATION

The main ideas of generating multiscat group are as follows.

- Source coding: Layered source streams are generated from a layered scalable codec where the number of layers and the bit rate of each layer can be adjusted by the sender. Let $V_h (h=1,...,H)$ represent primary streams. It can be seen that there is a linear dependency among $V_h$. That is, if a user subscribes $V_h$, one must subscribe all $V_j (1 \leq j < k)$ as well.
- Channel coding: This scheme doesn’t require specific channel-codes. However, the performance requires that for each stream, as long as the receiver has received any $K_h$ packets within total $N_h$ packets, the original $K_h$ source packets should be reconstructed in case of packet loss.
- Multicasting group generation: For given $K_h$ source packets, there are $N_h-K_h$ corresponding protection packets. We divide them into $F_h (F_h \leq N_h-K_h)$ protection streams. Each elementary stream is an independent multicasting group.
- Delay parameters: For each primary stream, different protection streams have different delays. Within a single protection stream, the delay can be regarded as a constant during a relatively short period, but may be changeable in a long run.

Note that some of the above processes are similar to the ones in [8].

Figure 3 shows an example to explain how it works. There are two primary streams $V_1$ and $V_2$. For $V_1$, (9,5) parity check is used and there are four protection streams $P_{11}-P_{14}$. The shadowed packets form a (9,5) group. For $V_2$, (6,4) parity check is used and there are two protection streams $P_{21}-P_{22}$.

3.1 Mathematical model for multicast parameter optimization

A receiver is described as a vector $U = (B_r, L_p)$, where $B_r$ is its available bandwidth or desired bit rate and $L_p$ is its packet loss ratio. Given the Rate-Distortion function in each video layer, the total number of layers and the channel coding efficiency, we can obtain an optimal rate distribution for this receiver which is described by a vector $(V_{1r}, ... , V_{Hr}, P_{1r}, ... , P_{pr})$. Notice that when $v_a = \sum_{k=1}^{N} v_k$ and $p_a = \sum_{k=1}^{N} p_k$, are given and so as R-D relations of different layers, the vector is fixed if it satisfies the minimal distortion requirement. Hence we can reduce the dimensions and represent the distribution as $D = (V_{ar}, P_{ar})$. We can describe the optimization scheme with a mapping $U$ to $D$: $D = G(U)$, where $G$ is known when the R-D relationships and the channel codec efficiency are known. We name the range of $U$ as $U$-space and that of $D$ as $D$-space. Furthermore, the receiver can be described as a random vector with PDF $f(U)$, in $U$-space. $f(D)$ indicates the probability of having a bandwidth of $B_r$ and packet loss ratio of $L_p$. In multicast case, from the sender’s point of view the combinative effect of all receivers corresponds to a combinative PDF in $U$-space.
space: 

\[ f_R(\mathbf{u}) = \frac{1}{M} \sum_{i=1}^{M} f_R(U_i) \]

where \( f_R(U_i) \) is the PDF for one receiver and \( M \) is the total number of receivers. If users have different priority, additional weights will be put into \( f_R(U_i) \). By using function \( G \), we can get its projection in \( D \)-Space: 

\[ f_D(D) = f_R(G^{-1}(D)) \]

Here we assume \( G^{-1} \) exists, which is usually physically valid.

Having known the \( R \)-Quality relation, we can further convert the value of \( V \) in \( D = (V, P) \) to an equivalent quality value \( Q \), i.e., \( f_R-D(V) = Q \). Thus, we get a new vector \( \mathbf{Q} = (Q, P) \) in \( Q \)-Space. The advantage of mapping to \( Q \)-Space is to pre-eliminate the effect of nonlinear \( R-D \) relationship in further process.

Before working on optimization, it is necessary for us to understand some properties of the \( D/Q \)-space and the \( D/Q \)-space coverage of a given multicast scheme. An example of the \( D \)-Space coverage of a receiver-driven layered multicast with FEC and pseudo-AQR is shown in Figure 5. Each star represents an exact amount of source data rate and protection that the scheme can provide. Change of parameters will change the locations of the stars, i.e., the \( D \)-Space coverage. Each dashed line represents a constant bandwidth, which corresponds to a horizontal line in \( U \)-Space. Another bunch of solid lines corresponds to constant packet loss ratios. If the \( R-D \) relation is linear and the same among all layers, they are straight lines as we draw, otherwise they are complicated curves. In this example, there are three primary streams, and each one corresponds a different number of protection streams. In practice, the protection bit rate is usually much smaller than the source data. Hence all stars in each horizontal line have a smaller bandwidth than those in a higher horizontal line.

In this section we will use dynamic programming and classification approach to solve parameter optimization. First, we determine the rate for each source group according to the classification of equivalent bandwidth. Secondly, the packet loss protection (on bit rate) for each group is determined.

3.2 A solution of the optimization by classification

In this section we will use dynamic programming and classification approach to solve parameter optimization. First, we determine the rate for each source group according to the classification of equivalent bandwidth. Secondly, the packet loss protection (on bit rate) for each group is determined.

- determine the data rate for each primary group
- determine the protection parameter for each group

Although a larger number of primary streams, \( H \), can achieve better performance, it will result in more network overhead for maintaining more multicast groups. Hence, we explicitly give a constraint of \( H_{max} \), which is a constant throughout the multicast session.

After getting the \( Q \)-Space coverage of all receivers, we project it onto the \( Q \)-axis. And then, we make the projection discrete, namely \( Q_h(i) \) \((i = 1, \ldots, L)\). Our problem now becomes to find \( Q_h \) \((0 < Q_h < \ldots < Q_h, h = 1, \ldots, H)\), the quality of video, when subscribing up to \( h \) layers to maximize the value of \( \sum_{i=1}^{L} \tilde{Q}_h(i) \), where \( \tilde{Q}_h(i) = Q_h \) when \( Q_h < Q_h(i) < Q_h \). It is easy to prove that for any \( h \), \( Q_h \) corresponds to a \( Q_h(i) \). An approximate solution is first to classify the projection data into \( H \) groups by \( K \)-means classification and choose the smallest \( Q \) value in each group (or nearly smallest value for reducing “outlier effect”), called \( Q_h \). Then, we locally move \( Q_h \) around among \( Q_h(i) \) one by one to get a better result. Note that when changing \( Q_h \), only users in user group \( h \) and some users in group \( h-1 \) would be affected.

IV. RECEIVER-DRIVEN FUNCTIONALITIES

The major work on the receiver side is to estimate the available bandwidth (desired bit rate), generate feedback without causing implosion, and efficiently arrange joining/leaving groups.

In our unicast rate control scheme [9], the available bandwidth (desired bit rate) is estimated using the formula derived in [6] by taking TCP-friendly property into consideration. In multicast case, some modifications are needed since the round trip time cannot be gotten directly. There are two categories of estimation schemes: (1) memoryless estimation and (2) estimation with memory. The first
one focuses on the relationship between network parameters; while the second one is based on the continuous property of bandwidth. In practice, people usually use a hybrid scheme of $\bigcirc$ and $\bigotimes$.

Our solution for avoiding feedback implosion is to use unicast as the feedback path. We reduce feedback rate when the number of receivers increase. Each receiver sets a timer with a random initial value coming from a uniform distribution from 0 to $X T$, where $X$ is an adjustable factor. Each time when the timer expires, it will send a feedback. The sender would give a control message to all receivers to inform them to increase the value of $X$ when there are too much feedbacks coming, and vice versa. Hence we can try to properly set $X$ value to reduce feedback traffic without affecting the sender making optimization.

Schemes for group joining/leaving without bandwidth estimation was introduced in [1, 2], although the adjustment of the initial value of joining test timers in [1] reflects some degree of estimation was introduced in [1, 2], although the adjustment of the sender making optimization.

The simulations are to show effectiveness of our sender adaptive scheme. We assume that there are 100 receivers in total, who belong to 5 bandwidth categories, ranging from 64kbps to 2048kbps. There are 5, 5, 20, 30, 40 users corresponding to the categories, respectively. In each category, the bandwidth probability density functions of all receivers follow the same Gaussian distribution. The mean values of the distribution vary from time to time, although the variances of distribution are fixed. In each simulation, we change the means of the Gaussian distribution randomly and the instant bandwidth of each receiver changes correspondingly. Rather than evaluate performance gain from the FEC and pseudo-AQR, which has been investigated in [8], we focus on examining the sender-adaptive case. We constantly compare the overall performance (the sum of received video qualities of all receivers and of all 500 time units per simulation) with the one when the sending data rates consistently equally distribute among the range between 64kbps to 2048kbps. The receiver-driven strategy is used in our simulation and the number of multicast groups is the same with our sender-adaptive case. “K-means” classification is used without further local adjustments.

Figure 6 shows performance evaluation in different type of situations. In Figure 6 (a), different curves correspond to different variance values of the Gaussian distributions. The variance reflects the degree of the bandwidth dynamics in each bandwidth category. It can be seen from Figure 6 (a) that the fewer the number of layers, the more the improvements. This is because when the number of the layers increases, the rates of each layer can fit individual receivers better even if the rates are evenly located. In addition, the higher the bandwidth dynamics of each user category (the closer the overall user distribution to a uniform shape), the less the improvements. In Figure 6 (b), different curves correspond to different Rate-Distortion relation. It can be seen that the performance improvements are R-D dependent. From Figure 6 (a) and (b), we can see a significant improvement in small layer-number cases. On the other hand, the computational cost of the sender optimization is very low.

VI. CONCLUSIONS

In this paper, first we proposed a novel scheme for video multicast over Internet, which is called sender-adaptive and receiver-driven layered multicast. This novel approach can be regarded as “a loose close loop”. Secondly, a general mathematical model for layered multicast to receivers with heterogeneous bandwidth and packet loss ratios has been introduced and a solution with explicit classification has been given. In our model, the different Rate-Distortion relationships and unequal error protection for different layers have been taken into account for the parameter optimizations. Note that our approach is at application layer only and no need to put merger in the router as in [4, 10]. Simulation results have shown that the sender-adaptive approach can achieve significant performance improvement. More efforts need to be made for developing a real experimental system for our scheme.

REFERENCES