

Toward Picture-perfect Streaming on the Internet*

Alix L.H. Chow
Computer Science Dept.
Univ. of Southern California
lhchow@cs.usc.edu

Leana Golubchik
CS & EE-Systems Depts, IMSC, ISI
Univ. of Southern California
leana@cs.usc.edu

John C.S. Lui
Dept. of Computer Science & Engineering
The Chinese Univ. of Hong Kong
cslui@cse.cuhk.edu.hk

Abstract

Quality of service (QoS) in streaming of continuous media over the Internet is poor, which is partly due to variations in delays, bandwidth limitations, and packet losses. Although continuous media applications can tolerate some missing data, non-recoverable information loss degrades these applications' QoS. Consequently, a number of application areas have backed away from streaming of their content over the Internet. Inability to control the resulting visual and auditory quality of the streamed presentation is an important reason for such a trend.

We believe that this trend can be reversed. To this end, this paper gives an overview of our efforts in exploring high quality streaming through the exploitation of multiple paths existing in the network. By high quality, we mean with significant bandwidth requirements, of relatively long duration, and without information loss or hiccups in data delivery. We believe this to be a promising approach.

1 Introduction

Quality of service (QoS) in streaming of continuous media (CM) over the Internet is poor, which is partly due to variations in delays, bandwidth limitations, and packet losses. Although CM applications can tolerate some missing data, non-recoverable information loss degrades these

applications' QoS. Consequently, a number of application areas (e.g., those related to the entertainment industry) have backed away from streaming of their content over the Internet. Inability to control the resulting visual and auditory quality of the streamed presentation is an important reason for such a trend.

We believe that this trend can be reversed. To this end, this paper gives an overview of our efforts in exploring high quality streaming through the exploitation of *multiple paths* existing in the network. By high quality, we mean with significant bandwidth requirements, of relatively long duration, and without information loss or hiccups in data delivery. In this paper, we will give a summary of our findings and briefly present evidence that multi-path streaming is a promising approach.

Our goal is to design an *application-level* approach, i.e., one that can be deployed over the Internet today, without requirements for support from/modifications to the lower layers of the network. This is in contrast, for instance, to an approach which would use the IntServ model for signaling (e.g., RSVP) and resource reservation in all routers along the streaming path. Such an approach would suffer from scalability and deployment problems.

Specifically, we investigate the potential benefits of providing QoS in CM delivery through the *exploitation of multiple paths* existing in the network between a set of senders and a receiver. One advantage of this approach is that the complexity of QoS provision can be pushed to the network edge (an original design principle of the Internet) and hence improve the scalability and deployment characteristics while at the same time provide a certain QoS level.

Our focus thus far has been on providing a fundamental understanding of the benefits of using *multiple paths* to deliver CM data (such as video) destined for a particular receiver, i.e., this data is fragmented into packets and the dif-

*This research has been funded in part by the NSF ANI-0070016, NSF EIA-0091474, NSF CCR0113192, RGC, as well as the Croucher Foundation Scholarship and the Okawa Research Award. It has also been funded in part by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, Cooperative Agreement No. EEC-9529152. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the National Science Foundation.

ferent packets take alternate routes to the receiver. This is illustrated in the example of Figure 1, where streaming over multiple paths is accomplished by streaming the data from three servers, distributed over a wide area network. In this example any server can send any fraction of the CM data; specifically, server i sends fraction α_i of the data expected by the receiver, where $0 \leq \alpha_i \leq 1$ and $\sum_i \alpha_i = 1$. This can be achieved by determining a sending pattern for each server, e.g., as in Figure 1, where each server only sends packets depicted by the solid rectangles. As each packet is sent by only one of the senders, the total amount of data sent is the same as in a single path case, i.e., our approach does not increase the overall workload on the network.

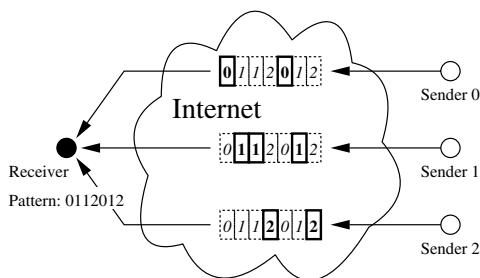


Figure 1. Multi-path streaming.

In general, we assume that the setting and possible adaptation of these fractions, as the delivery of data progresses, is done by the receiver, based on its perceived quality of data and determination of loss characteristics on the various paths corresponding to the servers. Thus, the receiver assembles the data from multiple senders and plays it in an appropriate order.

There are a number of approaches to accomplishing a multi-path data delivery. We later describe the specific approach considered in our work. We first note that such paths do not have to be completely disjoint, i.e., it is sufficient for them to have disjoint points of congestion or bottlenecks. Existence of multiple paths with *disjoint bottlenecks* includes the following potential benefits.

- *Reduction in correlations between consecutive packet losses.* Although a CM application can tolerate some missing information, a large number of consecutive packet losses not only contributes to significant degradation in CM quality but also diminishes ability to correct such losses through error correction techniques, e.g., erasure codes. As shown in [20], sending data through multiple paths can potentially reduce burst lengths and correlations between consecutive losses and thus improve the quality of delivered data.

- *Increased throughput.* In delivery of continuous media one can tradeoff the quality of the data with the amount of compression achieved, i.e., one can reduce the amount of bandwidth needed to deliver the data at the cost of its quality. Sending data through multiple paths potentially increases the amount of (aggregate) bandwidth available to the application and hence increases the quality of delivered data.
- *Improved robustness and ability to adjust to variations in congestion patterns.* CM applications are often long lasting (e.g., delivery of a movie might take on the order of hours). Hence, it is reasonable to expect that network conditions will change throughout the delivery of data to a CM application. Since not all paths, in general, would experience the same traffic patterns and congestion, sending data through multiple paths potentially improves the ability to adapt to changes in network conditions. Moreover, ability to use multiple paths can result in improved robustness in the face of failures.

Given these benefits, the use of multiple paths in distributed (over best-effort wide-area networks) CM applications, in general, requires consideration of the following research problems. (This is not an exhaustive list.)

- *Determining bottlenecks, joint points of congestion, and network characteristics in general.* To gain the benefits of multi-path streaming described above, one must first determine the paths to be used in delivery of the data. Since it is reasonable to characterize a path using its bottleneck link [12], what we need to be able to do is determine whether a number of paths share points of congestion, i.e., have joint or disjoint bottlenecks [21, 29]. In addition, it may be useful (for some approaches to multi-path streaming) to be able to estimate current capacities and loss characteristics of these bottlenecks. Although this has not been critical in our approach thus far, other approaches to multi-path streaming might require fairly accurate estimation of various network characteristics (refer to Section 3). These are non-trivial problems, but they are outside the scope of this discussion. We note that currently we use [29] in our system for detecting shared points of congestion.
- *Effects of redundancy and error erasure schemes.* Some amount of lost data can be reconstructed in CM applications through the use of redundant information, e.g., as in FEC [11] techniques. Hence, in constructing multi-path streaming techniques one should take into consideration how the erasure codes interact with multi-path delivery and the effect of redundant information on the final quality of the data. In our work, we

try to understand the benefits of multi-path streaming without focusing on the characteristics of a particular redundancy scheme.

- *Media coding scheme integration.* Sophisticated media coding techniques can be adopted to work closely with the multi-path streaming scheme (refer to Section 3 for a detailed discussion). In order to increase the flexibility of multi-path streaming deployment, we again try to gain understanding on the multi-path streaming benefits without focusing on a particular media coding scheme.
- *Load distribution.* As multiple paths are utilized between the set of senders and the receiver, one can assign different amounts of streaming workload to the different paths. The goal here would be to optimize the viewing quality of the resulting CM data and to perform the load distribution based on the quality of the various paths. This problem is explored in [1] and [18] under different path modeling methods. Example results are also presented in Section 2.
- *Adaptation schemes under changes in network conditions.* When network conditions change, one can improve the quality of CM data delivery by adapting the load distribution on the different paths (e.g., by sending less data on the more congested paths).
- *Data placement.* Proper placement of data on the servers is an issue in the context of CM applications delivering *pre-stored* data, e.g., a video-on-demand application (in contrast to a video conferencing application where data is produced “live”). Inappropriate data placement can adversely affect servers’ performance. For instance, this can occur due to load imbalance problems arising from the fact that only specific parts of the data are being delivered from a particular server as well as the fact that specific data required might change over the course of the application, as the system adapts to congestion patterns in the network. This in turn reduces the quality of service experienced by the CM application (in this case due to server rather than network performance). We note that these problems can be more severe when adaptation schemes (as mentioned above) are used.
- *Data dispersion.* Given that one cannot necessarily rely on the network layer to provide multi-path routing, another consideration is how to accomplish the dispersion of data over multiple paths existing in the network between a sender and a receiver of data. This may be an especially important consideration for applications where data is generated live, e.g., a video conferencing

application, where it may be necessary to use a collection of relay hosts or proxies to “force” paths different from those provided by the network between a sender and a receiver. In contrast, this may be less of an issue for applications where data is pre-recorded and can, for instance, be dispersed to a set of distributed servers in advance of actual data streaming.

- *Need for protocol/network support.* Lastly, some mechanisms for streaming data over multiple paths might require support from lower layers, such as the network layer. Of course, in this case, ease of deployment is an issue. (We do not require such support.)

From the above list, we can see that there are a number of difficult problems which need to be addressed when employing multi-path streaming. In Section 2, we give examples of results we obtained thus far, which indicate that this is a worth while approach to achieving high quality streaming over best-effort networks.

We note, however, that when one adopts multi-path streaming, its potential costs or detrimental effects should also be considered. For instance, MP streaming might have an adverse effect on the resulting delay characteristics observed at the receiver. As a result, it might also require additional amounts of receiver buffer space. In addition, the overheads associated with sending data over multiple paths and then assembling it into a single stream at the receiver should also be considered. Moreover, the overheads and complexity due to measurements needed to achieve better performance with MP streaming should also be considered. For instance, in our case, we employ detection of shared points of congestion [29]. Other approaches to MP streaming might require even more detailed information about the network (refer to Section 3) which is likely to result in a need for more “intrusive” and complex measurements. Lastly, scalability of such measurement schemes is an issue as well. However, the evaluation of such costs is outside the scope of this paper.

Lastly, although all the research issues discussed above are of importance, in our work thus far we have narrowed the scope by focusing on: (1) delivery of *pre-stored* video, e.g., as in video-on-demand applications (in contrast to delivery of “live” data); (2) *application-level* schemes (which are deployable today over the current Internet)¹; (3) accomplishment of multiple paths to the same receiver by *distributing servers* across wide-area networks and streaming data from multiple senders simultaneously; and (4) streaming over the network issues *only* (rather than, e.g., consid-

¹That is, we assume the use of best-effort IP-based networks, where a specific path is used between any pair of hosts (sender and receiver) on the network **and** this path is determined by a network-level routing algorithm; furthermore, our system does not require specific knowledge of the paths, only the ability to determine whether two paths share a point of congestion, e.g., by using [29].

ering server-related problems such as the load balancing issues mentioned above)².

2 Overview of Results

We now give a brief overview of our results thus far, to illustrate the potential of multi-path streaming in providing high quality streaming over best-effort networks.

2.1 Performance Metrics

We begin with a description of performance metrics considered in our work. Firstly, *loss rate*, P_N is the fraction of lost packets as seen by the receiver when one uses $N \geq 1$ paths for CM streaming. Secondly, *lag-1 autocorrelation*, the lag-1 autocorrelation function, $R[X(t)X(t + \delta)]$, measures the degree of dependency of consecutive packet losses as seen by the receiver, where $X(t)$ is a random variable indicating whether the packet sent at time t is lost or received properly (depending on the state of the GM) and $1/\delta$ is the bandwidth requirement (in units of packets/sec) of the streaming application³. Thirdly, *burst length of lost packets*, which is the probability mass function of consecutively lost packets as seen by the receiver. Note that if the lost packets burst length is large, it can (a) significantly affect the viewing quality of the CM object and (b) reduce the effectiveness of an error correction scheme, if some form of an erasure code is deployed. Lastly, when considering the use of erasure codes, we also consider *mean information loss rate (MILR)*, which accounts for loss of *media data* (i.e., not including loss of redundant packets). Although this may be a useful metric for indicating the resulting visual quality of CM data, it is often not easy to compute analytically.

2.2 Modeling and Analysis of Benefits

In order to first explore the benefits of multi-path streaming in a more analytical setting, we model the loss characteristics of a path⁴ using a stationary continuous time Gilbert model (GM) which can characterize the potential correlations between consecutive packet losses on a network path. For a GM, the packet loss process along path k is described by a two state continuous time Markov chain $\{X_k(t)\}$ where $X_k(t) \in \{0, 1\}$. If a packet is transmitted

²That is, for the purposes of this discussion we assume that the data is fully replicated at all servers and hence any server can deliver any fraction of the CM data.

³A high positive value of $R[X(t)X(t + \delta)]$ implies that a lost packet is very likely to be followed by another lost packet. A high negative value of $R[X(t)X(t + \delta)]$ implies that a lost packet is likely to be followed by a successful packet arrival. If the statistics of the consecutive packet losses are not correlated, then $R[X(t)X(t + \delta)] = 0$.

⁴Here we essentially assume that a path is characterized by its bottleneck link.

at time t when the state of path k is $X_k(t) = 0$, then the transmitted packet is received correctly by the receiver; the transmitted packet is considered lost if $X_k(t) = 1$.

Using the GM and the above suggested metrics, an initial work in [20] gave an analytical characterization of when a multi-path (MP) approach is beneficial, as compared to a single path (SP) approach. The results indicated that: (1) in general, multi-path streaming exhibits better loss characteristics than single-path streaming, (2) use of an erasure code may not necessarily improve data loss characteristics in the case of single-path streaming, while multi-path streaming (with or without use of an erasure code) can improve data loss characteristics, and (3) lag1-autocorrelation of multi-path streaming is usually closer to zero than that of single path streaming, which we believe should also result in a higher viewing quality of the received CM data. It was also indicated that the average error burst length in multi-path streaming is statistically shorter than that of single path streaming. As an example, consider Figures 2 and 3 which depict the error burst length distribution obtained from a prototype experiment⁵ using single path and multi-path streaming, respectively. These figures illustrate that multi-path streaming results in shorter error bursts; here, almost all bursts in multi-path streaming are one packet long.

2.3 Load Distribution

Given the above stated benefits, one important question that remains is how to distribute the load among multiple paths so as to optimize viewing quality. In [20] Golubchik et al. study MP streaming benefits using a *round-robin* (RR) approach to split the CM traffic among the multiple paths. In contrast, in [1] Golubchik et al. study the problem of optimally assigning CM traffic to the multiple paths, according to the path characteristics, such that a certain specified performance metric is optimized. This is done under the GM while considering both appropriate optimization objectives as well as the resulting streaming performance, all with a high level goal of improving the perceptual quality of the streamed media. We note that it is not trivial to pick an appropriate optimization objective for load distribution. If we simply use one of the first three metrics given in Section 2.1, undesirable effects might arise. For example, optimizing the lag-1 autocorrelation may result in higher loss rates observed at the receiver, when the paths are not homogeneous. Determining and gaining insight into suitable optimization objectives is therefore important. Since there is a fundamental tradeoff between the frequency of losses and the corresponding loss correlations, it is natural to consider an optimization objective which encompasses both metrics. There are a number of ways to include both in a single met-

⁵In this prototype, actual data streams are used in experiments; however, losses are modeled using the GM.

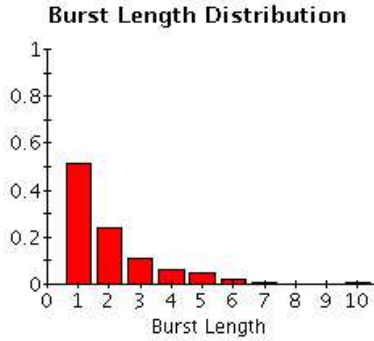


Figure 2. Single path prototype.

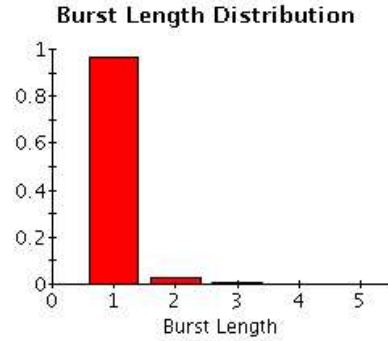


Figure 3. Dual path prototype.

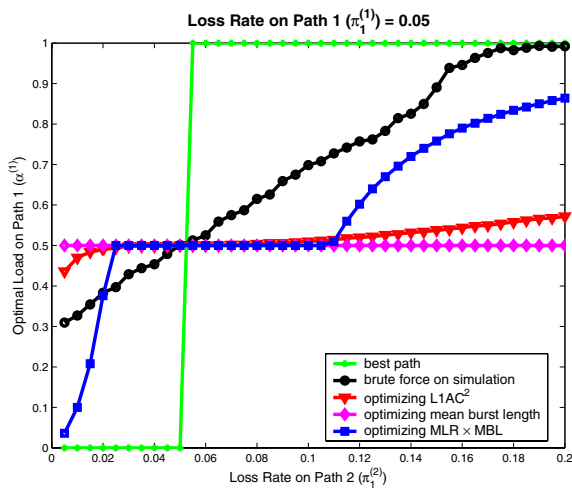


Figure 4. Optimal Load.

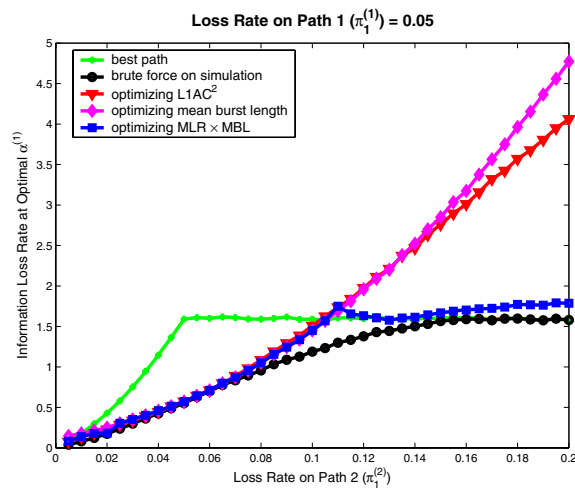


Figure 5. Information Loss Rate at Optimal.

ric. In [1], Mean Loss Rate (MLR) \times Mean Burst Length (MBL) is used as an example of such an optimization objective. Consider the corresponding example load distribution results⁶ [1] presented in Figure 4, which illustrates an optimal α_1^* as a function of loss rate on Path 2. (In this experiment, the Path 1 loss rate is 5% and the redundancy overhead (due to FEC) is 18.75%.) To compare the effects of different objective functions on α^* when FEC is used, an optimal traffic load distribution is also computed using MILR as the optimization objective by a *brute force* search on simulation results. (That is, all results other than those based on MILR are computed analytically here). The corresponding information loss rates, based on the load distribution results of Figure 4, are depicted in Figure 5. From

⁶Unless otherwise stated, MP streaming is performed on two paths with SP streaming using the “best path”, i.e., the one with the lower loss rate (the terms SP and Best Single Path streaming are used interchangeably). Each experiment is repeated 10 times with different random seeds, and the results are reported with $95\% \pm 10\%$ confidence intervals.

these figures and results based on other settings [1], the following observations can be made: (1) When Path 1 loss rate is relatively low (e.g., 5%), optimal load distribution on information loss increases relatively smoothly as a function of loss rate on Path 2; it eventually flattens out as it reaches the SP setting. (2) When Path 1 loss rate is low, optimizing on MLR \times MBL results in a load distribution closer to the optimal based on MILR. (3) When Path 1 loss rate is higher, e.g., 10%, the MILR-based optimal is closer to the load distribution given by SP streaming. (4) If a reasonable amount of FEC redundancy is used, optimizing on MLR \times MBL gives good performance, i.e., lower information loss rate, where an MLR \times MBL system operates closer to the MILR-based optimum. We note that when the path loss rate is relatively high and the corresponding amount of redundant information is insufficient, using SP streaming may result in better performance.

Another interesting experiment in [1] compares the ef-

Path 2 Loss	k=8	k=16	k=32	k=64	k=128
0.5% (i)	98.75%	40.63%	18.44%	9.22%	5.00%
0.5% (ii)	77.50%	41.88%	21.88%	13.75%	9.06%
0.5% (iii)	86.25%	38.75%	18.75%	9.38%	5.16%
5% (i)	$\geq 1000\%$	147.50%	58.75%	31.88%	19.92%
5% (ii)	97.50%	55.63%	31.25%	20.31%	14.14%
5% (iii)	97.50%	55.63%	31.25%	20.31%	14.14%
10% (i)	$\geq 1000\%$	147.50%	58.75%	31.88%	19.92%
10% (ii)	128.75%	68.75%	40.63%	26.72%	19.53%
10% (iii)	128.75%	68.75%	40.63%	26.72%	19.53%

Figure 6. Overhead needed to achieve information loss rate of $\leq 0.1\%$; Path 1 loss rate = 5%.

fect of redundant information on SP (*case i*) and MP streaming, where a simple round-robin approach (*case ii*) and the MBL \times MLR-based optimization approach (*case iii*) are used in the case of MP streaming. Suppose path 1 has a fixed loss rate of 5%, and three scenarios where Path 2 has 0.5%, 5%, and 10% loss rates are studied. For SP streaming, the better path (with lower loss rate) is chosen. The objective of this experiment is to find the minimum FEC overhead needed to achieve a required quality of service, which for this experiment we define as a mean information loss rate of $\leq 0.1\%$. (Other values have been tried, and the results are qualitatively similar.) Figure 6 gives the corresponding mean FEC overhead requirements⁷; these results indicate the following. Performance of MLR \times MBL is similar to that of RR when the difference in path loss rates of the two paths is not large. This is due to the fact that MLR \times MBL results in even traffic splitting in these cases, which is also shown in Figure 4. Moreover, SP, RR, and MLR \times MBL have different “best” performance ranges. For example, when path 2 loss rate is 0.5%, SP performs best (requires the lowest FEC overhead) when the FEC group size is larger ($k \geq 32$). For the same path loss rate, MLR \times MBL performs better than RR when we use $k \geq 16$ for FEC. When path 2 loss rate is 5% or 10%, MLR \times MBL (or RR) perform better than SP streaming unless the FEC group size is quite large.

2.4 A More Expressive Model

The above discussion illustrated that one can obtain improvements in various metrics by employing MP streaming. This was shown mainly under the GM; we refer to this model as a “conventional Gilbert model” in the remainder of the paper. One major limitation of using a conventional Gilbert model is that the loss process of a path is *independent* of the bandwidth requirements of the streaming appli-

⁷This is averaged over 10 experiments; the results are reported with $95\% \pm 11\%$ confidence intervals. For each scenario, different FEC group sizes are tested by varying k , the number of packets in a FEC group.

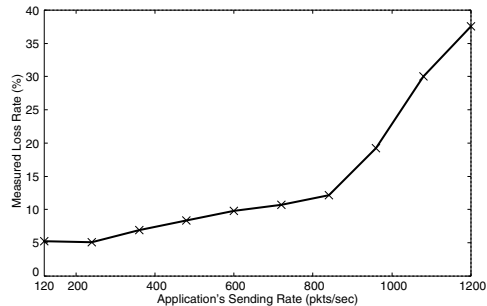


Figure 7. Data loss rate vs. sending rates.

cation — specifically, the loss rate as viewed by the receiver is fixed, independently of the sending rate of the application.

To illustrate the potential dependence of the loss rate on the application’s bandwidth requirements, consider the data loss rate as a function of an application’s sending rate depicted in Figure 7; this was obtained through an Internet experiment⁸. This figure illustrates the achieved loss rate for each experimental setting (i.e., *fraction* of lost packets as measured at the receiver), and it supports the hypothesis that a conventional Gilbert model may not be sufficient for characterizing the loss process of a path. Given this evidence, [18] proposes to use a *functional Gilbert model* (FGM) as a more general approach to characterizing the bursty loss nature of a path as well as its dependency on an application’s bandwidth requirements. Specifically, let λ denote an application’s average sending rate, in units of packets per second. For a stationary continuous time FGM, the packet loss process along path k is described by a two state continuous time Markov chain $\{X_k(t)\}$ where $X_k(t) \in \{0, 1\}$. Similarly to the conventional Gilbert model’s definition, if a packet is transmitted at time t when the state of path k is $X_k(t) = 0$, then no packet loss occurs; the transmitted packet is considered lost if $X_k(t) = 1$. The transition rate from state 0 to state 1 takes a *functional* form of $\mathcal{F}(\lambda)$. The transition rate from state 1 to state 0 also takes a *functional* form of $\mathcal{B}(\lambda)$. It is also assumed in [18] that $\mathcal{F}(\lambda)$ and $\mathcal{B}(\lambda)$ are continuous and furthermore that $\mathcal{F}(\lambda)$ is a *non-decreasing* function of λ and $\mathcal{B}(\lambda)$ is a *non-increasing* function of λ . Intuitively these assumptions make sense, and hence, in practice, they should not be restrictive.

⁸In this experiment UDP packets (1400 byte) were transmitted from Hong Kong to the USA, using a number of rates from 120 pkts/sec (around 1.34 Mbps) to 1200 pkts/sec (around 12.8 Mbps), with a step size of 120 pkts/sec interval. For each sending rate, the streaming experiment was carried out for 6 minutes, while measuring the corresponding achieved loss rate at the receiver. The experiment was carried out during daytime in Hong Kong, which corresponds to nighttime on the West Coast of the USA. Similar experiments were also performed using NS2 [22], and the results were qualitatively similar.

2.5 Load Distribution Revisited

We now revisit the load distribution problem under the FGM. Consider the following example result [18] with a 120 pkts/sec bandwidth requirement. We first consider a system with two heterogeneous paths wherein $\mathcal{F}_1(b) = 0.4 \times b$, $\mathcal{B}_1(b) = 21000/b$, $\mathcal{F}_2(b) = 0.0667 \times b$, and $\mathcal{B}_2(b) = 28500/b$, as an illustration. (Note that given the same packet rate on a path, path 2 has a better loss characteristic than path 1.) Under the *best* single path streaming approach (i.e., path 2 in this case), one can achieve a loss rate of 3.259%. Using 2-path streaming, we reduce the loss rate to 1.814%, with $\alpha^* = [0.260, 0.740]$.

We then add one more path to the system wherein $\mathcal{F}_3(b) = 0.233 \times b$ and $\mathcal{B}_3(b) = 24750/b$ (i.e., path 3 has loss characteristics in-between paths 1 and 2). With the addition of this path we reduce the loss rate further to 0.976%, with $\alpha^* = [0.190, 0.541, 0.269]$. This example illustrates that the traffic splitting flexibility of the multi-path approach provides us the opportunity to reduce the mean loss rate of a streaming application to a point which would not be possible with a single *best-path* type approach. Note that the additional path which was not present in the 2-path example is *not* the best of the three, yet it allows us to reduce the loss rate further.

The above results were obtained analytically and without the use of erasure codes. We now consider simulation results⁹ [18] where an error erasure code is used to reconstruct lost packets. Here the bandwidth requirements of the streaming application are increased by 12.5% due to the added overhead of the erasure code. Note that this overhead is the same for SP and MP streaming. The corresponding optimal load distributions are obtained based on the increased packet sending rate (due to the overhead).

The same three heterogeneous paths are considered, i.e., paths 1, 2, and 3 described above, where Table 1 illustrates the corresponding information loss rate for best path, two paths (with path 1 and 2), and three paths streaming. Load distribution among the paths is performed by optimizing packet loss rate as well as in a round-robin manner.

These results and further experiments given in [18] indicate that adoption of an erasure code, in most cases, can reduce the information loss rate. However, when the packet sending rate is high, employing an erasure code may have an adverse effect of increasing the loss rate (i.e., degrading the loss characteristics of paths). When an additional path is available, the workload (including the redundancy overhead) can be spread among paths; this results in better information loss rate. It is also observed that loss-based op-

⁹In these experiments, each data packet has a size of 1400 bytes. For each path, packet losses are emulated according to the FGM. Each simulation is analogous to a 24 hour media stream. The results are obtained using CSIM[23].

timal load distribution among the multiple paths can result in significantly better system performance than single best path streaming or the round-robin approach.

2.6 Experiments with Real Data

Chow et al. [18] discuss an experiment with real data using a multi-path multimedia streaming system prototype using the performance metrics given in Section 2.1 as well as the resulting visual quality under different load assignment methods. (A detailed description of the prototype can be found in [2].) Although packet losses are still emulated using the FGM, the MPEG stream and their processing is real, hence we are able to depict the resulting visual quality in addition to computing performance metrics¹⁰. Two senders, whose paths to the receiver have different loss characteristics, are used in this experiment. Path 1, which has better loss characteristics, is described by: $\mathcal{F}_1(b) = 0.0667 \times b$, $\mathcal{B}_1(b) = 28500/b$, and path 2, which has worse loss characteristics, is described by: $\mathcal{F}_2(b) = 0.4 \times b$, $\mathcal{B}_2(b) = 21000/b$, where $b = 192$ packets/sec. In this case, when one optimizes on the loss rate, $\alpha^* = [0.741, 0.259]$ is obtained.

Four cases with different traffic splitting vectors are studied: (Case 1) single path with *better* loss characteristics (using path 1), (Case 2) single path with *worse* loss characteristics (using path 2), (Case 3) dual path using *RR* traffic splitting, and (Case 4) dual path using *optimal* traffic splitting vector α^* . Packet loss statistics are measured at the receiver throughout the entire streaming process. Video frames are transcoded to JPEG files to allow a visual quality inspection.

Test Case	Loss Rate before FEC	Avg. Burst Length before FEC	Info. Loss Rate
1.	6.569%	1.762	1.688%
2.	35.630%	2.439	35.642%
3.	6.955%	1.034	1.000%
4.	3.403%	1.253	0.090%

Table 2. Prototype experiments.

Table 2 shows the average statistics measured for each test case. For the statistics before FEC, the statistics of the resultant packet receiving sequence after merging packets from different paths are measured. Figures 8(a), 8(b), and 8(c) depict the quality of the video frame sequence along playback time for Cases 1, 3, and 4, respectively¹¹. For each case, the transcoded video frame sequence is examined, and a vertical spike is depicted in the figure where

¹⁰A 150 seconds MPEG1 file which requires a playback rate of 174.5 Kbps (approximately 170 packets/sec with a packet size of 1024 bytes) is streamed. A FEC corresponding to an overhead of 12.5% is used, which increases the packet sending rate to around 192 packets/sec.

¹¹All video frames in Case 2 are damaged, and thus the results for that case are not presented.

Rate (pkts/sec)	2-path α^*	3-path α^*	2-path Opt. Info. Loss Rate	3-path Opt. Info. Loss Rate	Best SP Info. Loss Rate	2-path RR Info. Loss Rate	3-path RR Info. Loss Rate
60	[0.260,0.740]	[0.191,0.540,0.269]	0	0	0	0	0
120	[0.260,0.740]	[0.190,0.541,0.269]	0.001%	0	0.177%	0.057%	0
360	[0.260,0.740]	[0.190,0.541,0.269]	15.055%	5.301%	26.754%	26.128%	12.256%

Table 1. Optimization of load distribution revisited.

the video frame at that playback time is damaged (i.e., the picture exhibits a “blocking effect”). From these results, it

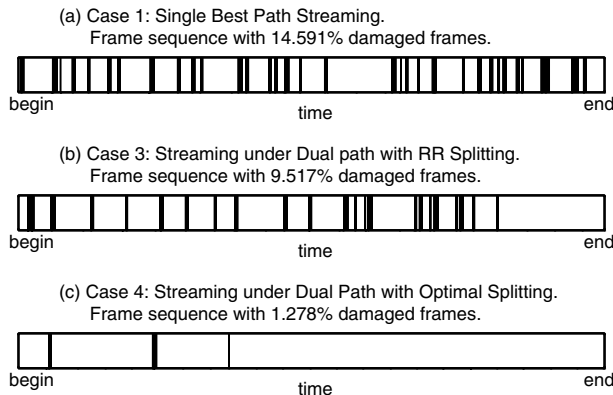


Figure 8. Video quality along playback time.

is observed that:

1. Loss rates (after FEC) significantly affect visual quality: When one relates the after FEC loss rate in Table 2 with the frame sequences quality indication in Figure 8, it is found that higher loss rate after the FEC operation, i.e., information loss rate, corresponds to poorer video quality, i.e., higher percentage of the video sequence is damaged. When the information loss rate is extremely high, for example 35.642% in Case 2, all the frames in the video sequence are damaged, i.e., the video is basically ruined and unviewable.
2. Introducing FEC may cause adverse effects: Improper adding of FEC may not improve the resulting information loss rate. In Case 2, it is found that loss rate before the FEC operation is less than the loss rate after the FEC operation. Adding FEC increases the loading along a congested path which may degrade the resulting loss characteristics and consequently increase data packet losses.
3. FEC performs better under multi-path streaming: As suggested in [20], multiple paths can reduce the lag-1 auto-correlation and shorten the error burst length; this enhances the error correction capability of FEC. The loss rates before FEC for Case 1 and Case 3 are 6.569% and 6.955%, respectively. Using the best path

in Case 1 can give a lower loss rate before the FEC process. However, after the error correction process, information loss is 1.688% for Case 1 and 1.000% for Case 3. The visual quality, which is strongly related to the information loss rate, is better in Case 3. It shows that simply using the best single path for video streaming may *not* be a good approach.

4. A path with worse loss characteristics can be used to improve the overall performance: Although path 2 is a worse path than path 1, it can still be used to share a fraction of the workload to improve the overall quality of the received video. From the statistics in Table 2 and the corresponding video output, the two multi-path test cases give better performance than the two single path test cases.
5. Using optimal traffic can result in better performance: In the experiments, when one assigns traffic loads according to an optimized traffic splitting vector, i.e., Case 4, the loss rate is the lowest both, before and after FEC. It shows that simply splitting the traffic evenly between paths may not result in the best use of available multiple paths (although it may lower the packet loss correlations).

3 Related Work

Multi-path streaming and exploitation of path diversity has attracted much attention recently, and [7] provides a broad overview of the general area. Here we give a brief survey of existing work on this topic focusing on those works which either consider loss characteristics or can be deployed over best-effort networks, as these are considerations in our work as well. As mentioned earlier, [20] illustrates the potential benefits of using multi-path streaming to improve the quality of the delivered CM data as compared to single path streaming. This is done by illustrating lower loss burst lengths and lower correlations in consecutive packet losses. A later work [1] studies the load distribution problem in multi-path streaming and shows that both the packet loss rate and the loss correlations are important when choosing an optimization objective. However, all these works are performed under the assumption that the application’s sending rate does not affect the loss rate on a path (i.e., the conventional Gilbert Model is used). In con-

trast, [18] illustrates the utility of considering an application's sending rate and the resulting effects on the loss characteristics of the CM streaming application.

Other works try to achieve multi-path streaming with assistance of the lower layers, e.g., [19] focuses a protocol which utilizes bandwidth available on multiple paths and redistributes workload according to congestion detected by a receiver. However, it requires network layer knowledge as well as a centralized routing server to compute the multi-path routing which minimizes the number of shared links while satisfying the bandwidth requirement. Given information about the underlying network graph, [17] proposes multi-path routing heuristics for unicast and multicast scenarios to achieve high bandwidth and low delay for video traffic. A data scheduling algorithm at the server is also proposed to minimize the end-to-end delay. Similarly, given network link information, [10] discusses a heuristic solution to find a set of paths that minimize the streaming distortion for a Multiple Description (MD) coded video stream. These works assume significant knowledge (i.e., link bandwidth and delay) and support (i.e., ability to control routing paths) from the underlying network. In contrast, our approach only deals with the end-hosts of the network and hence allows easy deployment on the Internet. Our only requirement is that chosen paths do not share points of congestion, which can be detected at the end-hosts using schemes such as [29]. Also, we focus on packet loss characteristics (e.g., loss correlations) rather than bandwidth and delay.

Sophisticated media coding techniques (e.g., MD coding) have been developed for use with multi-path streaming [4, 8, 6, 9, 5, 24, 30, 13, 14, 16, 15]. The basic idea is to partition the media into multiple roughly equally important independently decodeable bitstreams (descriptions). Each of them contains complementary information and is sent through different paths, such that media quality can be improved as the number of received descriptions increases. In this context, issues such as dealing with heterogeneous path bandwidth constraints [8], rate-distortion optimization [24, 13, 14], coding efficiency, adaptation schemes [30, 16], etc. are explored. Specifically, in [6, 9], a model based on the conventional GM is also used in the analysis and path selection for MD coded multi-path streaming. In our work, we focus on a more general study of multi-path streaming without the use of a specific coding technique - an advantage being that it can then be made to work well with many coding techniques or media codecs. Also, our work adopts a more expressive model for characterizing the loss characteristics of paths between senders and a receiver. Developing an adaptation scheme which reacts to time-varying path characteristics is our ongoing effort.

The work in [27] proposes the use of multi-servers for video streaming on the Internet. Their later work [26] extends the scheme by considering the use of FEC. In their

work, they focus on designing a receiver-driven transport protocol which includes (a) a rate allocation algorithm (i.e., how receiver split the traffic load onto multiple paths in order to minimize the probability of media packet loss by taking network bandwidth, channel characteristics and FEC parameters into account) and (b) a packet partitioning algorithm (which ensures non-overlapping packet sending and minimizes startup delay). In [25], they examine the case where the last mile connection is the bottleneck and employ multi-path streaming otherwise. In these proposals, conventional GM is adopted while a more expressive model is used in our work. Besides focusing on the loss rate, we also propose an optimization approach using other loss characteristics, e.g., lag-1 autocorrelation. Their later work [28] extends the idea of using FEC with path diversity on an overlay framework. The motivation is to emulate multiple sources by the use of an overlay network, which is similar to [3] but using multiple redundant paths simultaneously. They propose a heuristic scheme to select redundant paths. It is suggested that 10% of the network nodes participating in the overlay is enough for providing sufficient redundant paths. Our approach can also be implemented on an overlay structure as well as be made to achieving multiple paths by using relay nodes. In fact, we have a P2P prototype implementation of our approach [2].

Best-path type approaches have also been studied. For instance, [31, 32] perform path switching to select the best path by estimating the "goodness" of a path from the perspective of a video and a VoIP stream. However, they do not exploit the benefits of path diversity (e.g., reduced correlation in packet losses, workload distribution, etc.) and are similar to the best single path case discussed in our work.

4 Conclusions

In this paper we focused on multi-path streaming as an approach to providing high quality streaming over the Internet. We discussed evidence that such an approach has promise. Although much research still remains on this topic, we believe that this evidence indicates that there is hope for being able to stream continuous media over the Internet via the multi-path approach, with high quality and without huge costs.

To this end (and to verify our analytical and simulation results under real world conditions), we designed and built a P2P-based multi-path streaming prototype [2], currently being used for streaming video experiments between UMD, CUHK, and USC. We chose to use a P2P network as the supporting architecture for our prototype because of the parallelism between simultaneous downloads in P2P systems and multi-path streaming. Currently, we are working on Internet based experiments using this prototype.

References

- [1] B. Abdouni, W. C. Cheng, A. L. Chow, L. Golubchik, W.-J. Lee, and J. C. Lui. Multi-path streaming: Optimization and performance evaluation. In *SPIE Conference on Multimedia Computing and Networking (MMCN)*, volume 5680, pages 216–227, San Jose, California, January 2005.
- [2] B. Abdouni, W. C. Cheng, A. L. Chow, L. Golubchik, and J. C. Lui. Picture-perfect streaming over the internet: Is there hope? *IEEE Communication Magazine special issue on Proxy Support for Streaming on the Internet*, 42(8):72–79, August 2004.
- [3] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris. The case for resilient overlay networks. In *HotOS VIII*, Schloss Elmau, Germany, May 2001.
- [4] J. Apostolopoulos. Reliable video communication over lossy packet networks using multiple state encoding and path diversity. In *Visual Communications and Image Processing*, January 24–26, 2001.
- [5] J. Apostolopoulos, W. Tan, and S. Wee. Performance of a multiple description streaming media content delivery network. In *IEEE International Conference on Image Processing (ICIP)*, September 2002.
- [6] J. Apostolopoulos, W. Tan, S. Wee, and G. Wornell. Modeling path diversity for multiple description video communication. In *IEEE Inter. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2002.
- [7] J. Apostolopoulos and M. Trott. Path diversity for enhanced media streaming. *IEEE Communication Magazine special issue on Proxy Support for Streaming on the Internet*, 42(8):80–87, August 2004.
- [8] J. Apostolopoulos and S. Wee. Unbalanced multiple description video communication using path diversity. In *IEEE International Conference on Image Processing (ICIP)*, October 2001.
- [9] J. Apostolopoulos, T. Wong, W. Tan, and S. Wee. On multiple description streaming with content delivery networks. In *IEEE INFOCOM*, June 2002.
- [10] A. Begen, Y. Altunbasak, and O. Ergun. Fast heuristics for multi-path selection for multiple description encoded video streaming. In *IEEE Conference on Multimedia and Expo*, pages 517–520, Baltimore, MD, July 2003.
- [11] R. E. Blahut. *Theory and Practice of Error Control Codes*. Addison Wesley, January 1983.
- [12] J.-C. Bolot, S. Fosse-Parisis, and D. Towsley. Adaptive FEC-Based Error Control for Internet Telephony. In *INFOCOM*, 1999.
- [13] J. Chakareski and B. Girod. Rate-distortion optimized packet scheduling and routing for media streaming with path diversity. In *IEEE Data Compression Conference*, Snowbird, UT, April 2003.
- [14] J. Chakareski and B. Girod. Server diversity in rate-distortion optimized media streaming. In *IEEE International Conference on Image Processing*, Barcelona, Spain, September 2003.
- [15] J. Chakareski, S. Han, and B. Girod. Layered coding vs. multiple descriptions for video streaming over multiple paths. In *ACM Multimedia*, Berkeley, CA, November 2003.
- [16] J. Chakareski, E. Setton, and B. Girod. Video streaming with diversity. In *IEEE Conference on Multimedia and Expo*, pages 9–12, Baltimore, MD, July 2003.
- [17] J.-C. Chen, S.-H. Chan, and V. Li. Multipath routing for video delivery over bandwidth-limited networks. *IEEE Journal on Selected Areas in Communications special issue on Design, Implementation and Analysis of Communication Protocols*, 22(10):1920–1932, December 2004.
- [18] A. L. Chow, L. Golubchik, J. C. Lui, and A. W.-J. Lee. Multi-path Streaming: Optimization of Load Distribution. Accepted for publication in *Journal of Performance Evaluation*, 2005.
- [19] H. Chu and K. Nahrstedt. Dynamic multi-path communication for video traffic. In *Hawaiian International Conference on System Science*, Hawaii, January 1997.
- [20] L. Golubchik, J. C. Lui, T. F. Tung, A. L. Chow, W.-J. Lee, G. Franceschinis, and C. Anglano. Multi-path Continuous Media Streaming: What are the Benefits? *Performance Evaluation*, 49:429–449, September 2002.
- [21] K. Harfoush, A. Bestavros, and J. Byers. Robust identification of shared losses using end-to-end unicast probes. In *The 6th IEEE International Conference on Network Protocols (ICNP)*, Osaka, Japan, October, 2000.
- [22] <http://www.isi.edu/nsnam/ns/>. *The Network Simulator - ns-2*.
- [23] <http://www.mesquite.com/>. *CSIM18*.
- [24] Y. J. Liang, E. Setton, and B. Girod. Channel-adaptive video streaming using packet path diversity and rate-distortion optimized reference picture selection. In *IEEE Fifth Workshop on Multimedia Signal Processing*, St. Thomas, Virgin Islands, December 2002.
- [25] T. Nguyen, P. Mehra, and A. Zakhor. Path diversity and bandwidth allocation for multimedia streaming. In *ICME*, Baltimore, MD, USA, July 6–9, 2003.
- [26] T. Nguyen and A. Zakhor. Multiple sender distributed video streaming. *IEEE Tran Multimedia*, pages 315–326, April 2004.
- [27] T. Nguyen and A. Zakhor. Distributed video streaming over internet. In *SPIE Conference on Multimedia Computing and Networking*, San Jose, California, January 2002.
- [28] T. Nguyen and A. Zakhor. Path diversity with forward error correction (pdf) system for delay sensitive applications over the internet. In *IEEE InfoCom*, San Francisco, California, March 2003.
- [29] D. Rubenstein, J. Kurose, and D. Towsley. Detecting shared congestion of flows via end-to-end measurement. *IEEE/ACM Transactions on Networking*, 10(3):381 – 395, June 2002.
- [30] E. Setton, Y. Liang, and B. Girod. Adaptive multiple description video streaming over multiple channels with active probing. In *IEEE Conference on Multimedia and Expo*, Baltimore, MD, July 2003.
- [31] S. Tao and R. Guerin. Application-specific path switching: A case study for streaming video. In *ACM Multimedia*, New York, October 2004.
- [32] S. Tao, K. Xu, A. Estepa, T. Fei, L. Gao, R. Guerin, J. Kurose, D. Towsley, and Z.-L. Zhang. Improving voip quality through path switching. In *IEEE InfoCom*, Miami, March 2005.