Multi-path Streaming: Optimization and Evaluation

B. Abdouni* W. C. Cheng† A. L.H. Chow‡ L. Golubchik§ W.-J. Lee ¶
J. C.S. Lui||

Abstract

Quality of service (QoS) in delivery of continuous media (CM) over the Internet is still relatively poor, which is largely a result of packet losses often due to congestion. Previous work has shown that use of multiple paths, existing in the network between a set of senders and a receiver, to deliver CM can be beneficial and should lead to improved QoS. In this paper, we study the problem of load distribution, i.e., properly distributing a CM stream among the multiple paths. We first focus on determining an appropriate optimization objective for computing the load distribution. We then conduct a performance study to understand the goodness of our optimization objectives and the resulting benefits of using multiple paths to deliver CM data.

1 Introduction

Quality of service (QoS) in streaming of continuous media (CM) over the Internet is still poor and inconsistent. The degradation in quality of CM applications, involving delivery of video and audio, is partly due to variations in delays, bandwidth limitations, as well as losses experienced by packets sent through wide-area networks. Although many such applications can continue to operate with some degree of missing data, non-recoverable information loss degrades these applications' quality of service. 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.

One approach to providing QoS for CM applications over the Internet is to use the IntServ model for signaling (e.g., RSVP) and resource reservation in all routers along the streaming path. However, this approach suffers from scalability and deployment problems. In contrast, in this work we focus on providing QoS guarantees in CM delivery through the exploitation of *multiple paths* existing in the network between a sender (or a set of senders) and a receiver. The basic idea is that CM data destined for a particular receiver is fragmented into packets and the different packets take alternate routes to the 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 level of QoS guarantees. There are many other advantages of multi-path (MP) streaming over the Internet, some of which have been exposed by previous works, as described in Section 4.

*Department of Electrical Engineering Systems, University of Southern California, abdouni@usc.edu.
†Department of Computer Science, University of Southern California, bill.cheng@usc.edu.
‡Department of Computer Science, University of Southern California, lhchow@cs.usc.edu.
§Department of Computer Science, EE-Systems, IMSC, ISI, University of Southern California, leana@cs.usc.edu.
¶Department of Computer Science, University of Maryland at College Park, adamlee@cs.umd.edu.
||Department of Computer Science & Engineering, The Chinese University of Hong Kong, cslui@cse.cuhk.edu.hk.

There are a number of approaches to accomplishing multi-path data delivery; the specific approach considered in this paper is described in Section 2.1. We note that such paths do not have to be completely disjoint — it is sufficient for them to have **disjoint bottlenecks**. Existence of multiple paths with *disjoint bottlenecks* can result in many benefits (as detailed in [7]) including (a) increased bandwidth, (b) improved loss characteristics, and (c) ability to better adjust to changes in network conditions. In this paper, we focus on the loss characteristics only.

We also note that in this paper, we narrow the scope by considering delivery of *pre-stored* CM, *application-level* schemes which are deployable today over the current Internet (without changes to the network), accomplishment of multiple paths to the same receiver by *distributing servers* across wide-area networks and streaming data from multiple senders simultaneously, and streaming over the network issues *only*, rather than, e.g., considering server-related problems such as data placement issues.

To reap the benefits of multi-path streaming one must first consider and solve a number of research problems (as detailed in [7]). One such problem is the appropriate *load distribution* among the multiple path. *Optimum load distribution in the context of multi-path streaming is the focus of this paper.* We define this problem in more detail and more formally in Section 2.

The contributions of this work are as follows. We first focus on determining an appropriate optimization objective for computing an optimal load distribution, where the high level goal is to improve the perceptual quality of the resulting CM. We then conduct a performance study to understand the goodness of our optimization objective as well as gain insight into the resulting benefits of using multiple paths to deliver CM data. The contributions of this work as compared to the existing literature is discussed in Section 4.

Lastly, we note that one should also consider the potential costs or detrimental effects of multi-path (MP) streaming. For instance, MP streaming might have an adverse effect on the resulting delay characteristics observed at the receiver. As a result, it might require a large amount of receiver buffer space. Also, a simple implementation of MP streaming would require replicas of the CM data to be distributed to the servers (although, mirroring servers is a common practice). Advances in data placement approaches are needed to reduce such costs. 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 need to detect shared points of congestion, e.g., using [18]. Other approaches to MP streaming might require even more detailed information about the network which is likely to result in a need for more "intrusive" and complex measurements (refer to Section 4). Lastly, scalability of such measurement schemes is an issue as well. A quantitative evaluation of such costs is outside the scope of this paper.

2 Load Distribution Problem

In this section we formulate the optimum load distribution problem and analyze a set of possible corresponding optimization objectives.

2.1 Background

Consider streaming of pre-stored CM data from N senders, where sender i sends fraction $\alpha^{(i)}$ of the data expected by the receiver. The streaming traffic is split among these N senders according to a traffic splitting vector $\alpha = [\alpha^{(1)}, \alpha^{(2)}, \ldots, \alpha^{(i)}, \ldots, \alpha^{(N)}]$, where $\alpha^{(i)} \geq 0$ and $\sum_i^N \alpha^{(i)} = 1$. Assume the original packet sending rate, when using a single path is λ (in units of packets/sec). In our multi-path streaming scheme, sender k streams data at the rate of $\lambda^{(k)} = \alpha^{(k)}\lambda$, and $\delta^{(k)} = 1/\lambda^{(k)}$ represents the time interval between consecutive packets from sender k.

We assume that the N senders are chosen such that there are N paths between the senders and the client with **disjoint bottlenecks**. Moreover, we follow the work in [2, 7] and use a stationary continuous time Gilbert model (GM) to characterize the potential correlations between consecutive packet losses on *each* network path. That is, each path is characterized by its own GM. Since it is reasonable to characterize a path using its bottleneck link [2], what we need to be able to do is determine whether one path shares a point of congestion with another path and then choose paths for MP streaming which do not share bottlenecks. This can be done using previously proposed approaches, such as in [8, 18]. Since the paths are *disjoint*, the GMs are assumed to be *independent*.

For a stationary continuous time Gilbert model, 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 at time t when the state of path k is $X^{(k)}(t) = 0$, then the transmitted packet is received correctly by the receiver. On the other hand, the transmitted packet is considered lost if $X^{(k)}(t) = 1$. The infinitesimal generator for the Gilbert model characterizing path k is:

$$\boldsymbol{Q}^{(k)} = \begin{bmatrix} -\mu_0^{(k)} & \mu_0^{(k)} \\ \mu_1^{(k)} & -\mu_1^{(k)} \end{bmatrix}$$

where $\mu_0^{(k)}$ is the transition rate from state 0 to state 1 and $\mu_1^{(k)}$ is the transition rate from state 1 to state 0 corresponding to path k. The stationary distribution of this Gilbert model is $\boldsymbol{\pi}^{(k)} = [\pi_0^{(k)}, \pi_1^{(k)}]$ where $\pi_0^{(k)} = \frac{\mu_1^{(k)}}{\mu_0^{(k)}+\mu_1^{(k)}}$ and $\pi_1^{(k)} = \frac{\mu_0^{(k)}}{\mu_0^{(k)}+\mu_1^{(k)}}$.

Let $p_{i,j}^{(k)}(\tau)$ be the probability that path k is in state j at time $t+\tau$, given that it was in state i at time t, i.e., $p_{i,j}^{(k)}(\tau) = P(X^{(k)}(t+\tau) = j | X^{(k)}(t) = i)$. From [13], we have that

$$p_{i,j}^{(k)}(\tau) = \begin{cases} \frac{\mu_1^{(k)}}{\mu_0^{(k)}+\mu_1^{(k)}}\left(1 - e^{-[\mu_0^{(k)}+\mu_1^{(k)}]\tau}\right) & i = 1, j = 0, \\ \frac{\mu_0^{(k)}}{\mu_0^{(k)}+\mu_1^{(k)}}\left(1 - e^{-[\mu_0^{(k)}+\mu_1^{(k)}]\tau}\right) & i = 0, j = 1, \\ \frac{\mu_0^{(k)}+\mu_1^{(k)} e^{-(\mu_0^{(k)}+\mu_1^{(k)})\tau}}{\mu_0^{(k)}+\mu_1^{(k)}} & i = 1, j = 1, \\ \frac{\mu_1^{(k)}+\mu_1^{(k)} e^{-(\mu_0^{(k)}+\mu_1^{(k)})\tau}}{\mu_0^{(k)}+\mu_1^{(k)}} & i = 0, j = 0 \end{cases}$$

for all $\tau > 0$.

In [7], the authors study and compare the performance of single path (SP) streaming vs. multi-path (MP) streaming using the Gilbert model and the following metrics:

- **Loss rate:** P_N is the fraction of lost packets as seen by the receiver when one uses $N \geq 1$ paths for streaming.
- **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 Gilbert model) and $\delta = 1/\lambda$. For instance, a 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. If the statistics of the consecutive packet losses are not correlated, then $R[X(t)X(t+\delta)] = 0$.

- **Conditional burst length of lost packets:** this is the probability mass function of consecutively lost packets, conditioned on there being a burst of packet loss, as seen by the receiver. Intuitively, long bursts will significantly affect the viewing quality of the multimedia object. Moreover, they will reduce the effectiveness of an error correction scheme, if some form of an erasure code is deployed.

In [7], the potential benefits of using MP streaming are analyzed using the above given GM and the above given metrics. To motivate the importance of considering loss correlations, we include in the Appendix A an experiment showing how correlated bursty losses affect video quality.

2.2 Motivation

In [7] MP streaming benefits are studied using a *round-robin* approach to splitting the CM traffic among the multiple paths. In contrast, here we focus on 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. That is, given the Gilbert models for the N streaming paths, we compute the optimal traffic splitting vector, α^* according to an optimization objective, where $\sum_i^N \alpha^{(i)} = 1$. We refer to this as the *load distribution problem*, and we study 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. If we simply use one of the metrics mentioned above, 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 an important part of this work. In order to obtain a basic understanding of the problem, we first study the resulting performance in an analytical setting, without the use of erasure codes. Subsequently, we present simulation results with erasure codes as well.

2.3 Optimization Objectives

In order to attain desirable performance properties of multi-path streaming, appropriate load distribution, α, needs to be chosen, as the load distribution will affect the quality of the streamed media. In this paper, we will approximate media quality using information loss rate (as defined in Section 3). Since most streams are expected to be transmitted with some form of an erasure code[1], the information loss rate will be a function of the erasure code used as well. However, the analysis below first focuses on a setting without an erasure code, in order to gain insight into the problem independently of a particular error correcting scheme.

Of course, choosing the optimal α is a factor of the optimization objective. The resulting quality of the stream, however, is a factor of how well that optimization objective approximates media quality under packet losses. Since the authors in [7] propose to describe stream quality using the three metrics stated above, we begin with those as our optimization objectives. We evaluate the considered optimization objectives under the above described Gilbert model, using two paths.

[1] Here we do not consider retransmissions, partly because there is usually little opportunity to retransmit data in such real-time applications, and hence some amount of lossiness may have to be tolerated. In a complete system a hybrid approach (with retransmissions and erasure codes) might be useful. However, this paper is focused on exploring loss correlations which have a more significant effect on error correcting schemes; we believe that adding retransmission to our evaluation process would obscure the characteristics being illustrated here.

2.3.1 Mean Loss Rate

Since our approximation of quality here is the mean information loss rate, it is natural to start our search for the best optimization objective with the Mean Loss Rate (MLR). The mean loss rate, ($\bar{\mathcal{X}}$), observed at the receiver when streaming over multiple paths is given by $\bar{\mathcal{X}} = \Pr[\text{loss}] = \alpha^{(1)}\pi_1^{(1)} + \alpha^{(2)}\pi_2^{(2)}$. That is, optimizing on MLR is equivalent to streaming over the best (in terms of loss rate) single path (in what follows we use the terms interchangeably). And, as illustrated in Section 3, this does not always result in the best streaming quality — this does not allow us to take advantage of the reduced correlations in packet losses achieved when using multiple paths (as illustrate in [7]). Reducing such correlations (i.e., packet loss burst lengths) is important when erasure codes are applied — intuitively, the more losses are "spread out", the better the performance of an erasure code.

2.3.2 Conditional Mean Burst Length

Motivated by the need to consider burst lengths, we examine the conditional Mean Burst Length (MBL) next (conditioned on there being a loss) as an optimization objective. The conditional mean burst length, ($\bar{\mathcal{B}}$), is derived as:

$$\bar{\mathcal{B}} = \frac{\bar{\mathcal{X}}}{\Pr[\text{burst}]} = \begin{cases} \frac{[\alpha^{(1)}\pi_1^{(1)}+\alpha^{(2)}\pi_1^{(2)}]}{[\alpha^{(2)}-\alpha^{(1)}]\pi_0^{(2)}p_{0,1}^{(2)}(\delta^{(2)})+\alpha^{(1)}\pi_0^{(1)}\pi_1^{(2)}+\alpha^{(1)}\pi_0^{(2)}\pi_1^{(1)}} & \alpha^{(1)} < \alpha^{(2)} \\ \frac{\frac{1}{2}[\pi_1^{(1)}+\pi_1^{(2)}]}{\pi_0^{(1)}\pi_1^{(2)}+\pi_0^{(2)}\pi_1^{(1)}} & \alpha^{(1)} = \alpha^{(2)} = \frac{1}{2} \\ \frac{[\alpha^{(1)}\pi_1^{(1)}+\alpha^{(2)}\pi_1^{(2)}]}{[\alpha^{(1)}-\alpha^{(2)}]\pi_0^{(1)}p_{0,1}^{(1)}(\delta^{(1)})+\alpha^{(2)}\pi_0^{(1)}\pi_1^{(2)}+\alpha^{(2)}\pi_0^{(2)}\pi_1^{(1)}} & \alpha^{(1)} > \alpha^{(2)} \end{cases}$$

In our experiments, optimizing on MBL resulted in a round-robin approach. And, as illustrated in Section 3, this does not always result in the best streaming quality, e.g., this can occur when one of the paths has a high (relative) loss rate. In such cases, the losses due to "equal use" of highly heterogeneous paths become the dominant factor, i.e., the reduced correlations achieved by optimizing on MBL are not sufficient to overcome the corresponding losses. Hence, there is a fundamental tradeoff between mean losses and the corresponding loss correlations — this tradeoff is explored further below.

2.3.3 Lag-1 AutoCorrelation

Still motivated by the need to reduce loss correlations in order to improve the ability to correct losses through erasure codes, we consider the lag-1 autocorrelation (L1AC) as the optimization objective, which is derived as:

$$R[\mathcal{X}_t\mathcal{X}_{t+\delta}] = 1 - \frac{\Pr[\text{burst}]}{\bar{\mathcal{X}}(1-\bar{\mathcal{X}})}$$

$$\Pr[\text{burst}] = \begin{cases} [\alpha^{(2)}-\alpha^{(1)}]\pi_0^{(2)}p_{0,1}^{(2)}(\delta^{(2)}) + \alpha^{(1)}\pi_0^{(1)}\pi_1^{(2)} + \alpha^{(1)}\pi_0^{(2)}\pi_1^{(1)} & \alpha^{(1)} < \alpha^{(2)} \\ \frac{1}{2}\pi_0^{(1)}\pi_1^{(2)} + \frac{1}{2}\pi_0^{(2)}\pi_1^{(1)} & \alpha^{(1)} = \alpha^{(2)} = \frac{1}{2} \\ [\alpha^{(1)}-\alpha^{(2)}]\pi_0^{(1)}p_{0,1}^{(1)}(\delta^{(1)}) + \alpha^{(2)}\pi_0^{(1)}\pi_1^{(2)} + \alpha^{(2)}\pi_0^{(2)}\pi_1^{(1)} & \alpha^{(1)} > \alpha^{(2)} \end{cases}$$

under the assumption of a specific class of sending patterns, i.e., packets are distributed among paths such that no two consecutive packets are sent on the less loaded path. (In our experiments, such as sending pattern gave lower correlation than others.)

As illustrated in Section 3, optimizing on L1AC[2] Optimizing on L1AC expresses the need to consider correlations in packet losses better than optimizing on MBL. However, it suffers from the same shortcomings

[2] More specifically, we optimize on the square of lag-1 autocorrelation, so as to deal with positive values.

pointed out in the context of MBL above, when paths are highly non-homogeneous. Another problem with L1AC is that it is a very sensitive metric, and hence may not lead to robust load distribution approaches.

2.3.4 Mean Loss Rate × Mean Burst Length

Since there is a fundamental tradeoff between the frequency of losses and the corresponding loss correlations, as described above, we naturally consider an optimization objective which encompasses both metrics, i.e., we consider MLR × MBL as our final optimization objective, as derived below:

$$\bar{\mathcal{X}} \cdot \bar{\mathcal{B}} \;=\; \bar{\mathcal{X}} \cdot \frac{\bar{\mathcal{X}}}{\Pr[\text{burst}]} = \begin{cases} \frac{[\alpha^{(1)}\pi_1^{(1)}+\alpha^{(2)}\pi_1^{(2)}]^2}{[\alpha^{(2)}-\alpha^{(1)}]\pi_0^{(2)}p_{0,1}^{(2)}(\delta^{(2)})+\alpha^{(1)}\pi_0^{(1)}\pi_1^{(2)}+\alpha^{(1)}\pi_0^{(2)}\pi_1^{(1)}} & \alpha^{(1)} < \alpha^{(2)} \\ \frac{\frac{1}{2}[\pi_1^{(1)}+\pi_1^{(2)}]^2}{\pi_0^{(1)}\pi_1^{(2)}+\pi_0^{(2)}\pi_1^{(1)}} & \alpha^{(1)} = \alpha^{(2)} = \frac{1}{2} \\ \frac{[\alpha^{(1)}\pi_1^{(1)}+\alpha^{(2)}\pi_1^{(2)}]^2}{[\alpha^{(1)}-\alpha^{(2)}]\pi_0^{(1)}p_{0,1}^{(1)}(\delta^{(1)})+\alpha^{(2)}\pi_0^{(1)}\pi_1^{(2)}+\alpha^{(2)}\pi_0^{(2)}\pi_1^{(1)}} & \alpha^{(1)} > \alpha^{(2)} \end{cases}$$

We evaluate the goodness of these optimization objectives and the resulting performance of multi-path streaming next.

3 Performance Study

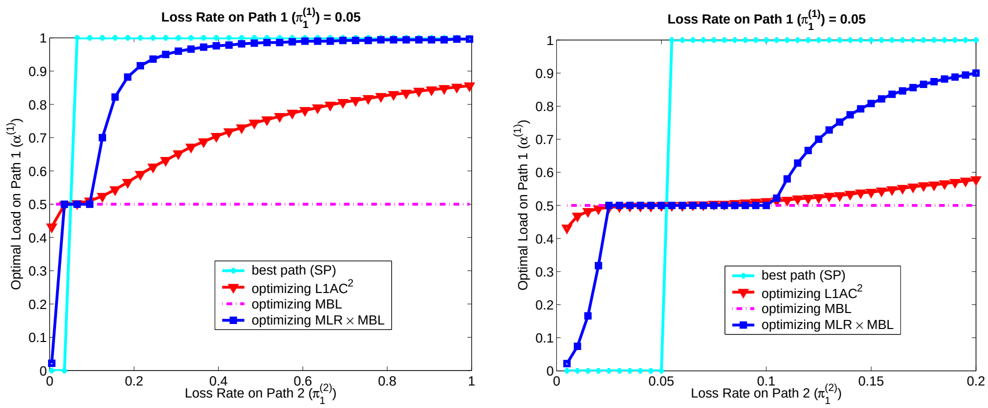
In this section, we present a performance study of SP and MP streaming under our load distribution optimization techniques. Results without the use of FEC are discussed first. Unless otherwise stated, the experiments below perform MP streaming on two paths with SP streaming using the "best path", i.e., that with the lower loss rate (we will use the terms SP and Best Single Path interchangeably). Given a desired path mean loss rate, we achieve it by fixing $\mu_1(i) = 70$ and varying $\mu_0(i)$ in the Gilbert model accordingly. The loss rate for Path 1 is set to 5% by default. Table 1 gives the default settings for the experiments. Results in Experiments 1 and 2 are computed using the analytical results of Section 2.

Table 1: Default Experimental Settings

Path 1 Loss Rate	5%
Path 2 Loss Rate	Varies
FEC k (except Exp 1 & 2)	32
FEC n (except Exp 1 & 2)	38

Experiment 1: Effect on Optimal Load Distribution: Here we study the optimal traffic distribution, α^*, resulting from using the objective functions proposed in Section 2. Figure 1(a) shows the optimal load assignment on Path 1 α_1^* , as a function of loss rate on Path 2. Each curve corresponds to a different objective function. Figure 1(b) shows a smaller segment of the results (where Path 2 loss rate is $\leq$ 20%) for cleairty of presentation. From these results, we make the following observations:

- The objective functions considered in Section 2 can produce significantly different α^*s.
- Optimizing by minimizing mean burst length results in a distribution where each path carries 50% of the load no matter how *different* in terms of loss rates Path 1 and Path 2 are, which can be implemented using a simple Round-Robin approach. (This turns out to be the case in all our experiments.)
- When optimizing MLR*MBL, α^* remains constant when paths are close to being homogeneous in terms of loss rates. This can be explained as follows. When the paths are nearly homogeneous, the MLR factor has a smaller effect and hence the optimization is dominated by the MBL factor. This results in a flat region which behaves similarly to optimizing on MBL alone. This result supports the intuition that when paths are nearly homogeneous, a simple Round-Robin approach should work well.



(a) (b)

Figure 1: Optimal Load Distribution for Different Objective Functions (without FEC)

- Optimizing the Lag 1 autocorrelation results in α^* less responsive to changes in loss rates, as compared to optimizing on MLR*MBR. This insensitivity can lead to high loss rates, which is illustrated in the next experiment.

Lastly, we can also observe that the more interesting results occur when Path 2 loss rate is reasonably low, e.g, $\leq 20\%$ in our experiments. This is reasonable as the loss rate on Path 1 is relatively low as well. Since the lower loss rates are reasonable to consider, given the current Internet, due to space limitations and for clarity of presentation, in what follows we focus on the segments of results with low Path 2 loss rates. (Please refer to the Appendix B for results showing the entire range of Path 2 loss rates.)

Experiment 2: Effect on Performance Metrics: Using the previous experiment's setup, we now investigate effects of the objective functions on several performance metrics of interest. Figures 2 and 3 show the resulting mean loss rate and mean burst length, respectively. Due to lack of space, Lag 1 autocorrelation results are included in the Appendix B, in Figure 13. From these results, we make the following observations:

- Optimizing on MLR*MBR may result in higher loss rates as compared to the Best Single Path. The maximum observed difference (around 3%) occured when Path 2 loss rate was 10%. However, using the Best Single Path gives significantly poorer performance in other metrics such as the mean burst length. We believe this will result in poorer perceptual quality; we elaborate on this below.

- For most metrics, performance obtained from optimizing on MLR*MBR stays between the results obtained from SP streaming and results obtained by optimizing on Lag 1 autocorrelation. While SP results in the lowest loss rate and optimizing on Lag 1 autocorrelation results in shorter mean burst lengths, optimizing MLR*MBL results in a reasonable tradeoff between these two metrics, both of which have an effect on the perceptual quality.

In the following experiments, we study the effects of using FEC on SP and MP streaming. Unless otherwise stated, we set FEC parameters to $k = 32$, $n = 38$, i.e., every 32 media packets are grouped together to generate 6 redundant packets. Thus, the FEC overhead is $\approx 20\%$. CSIM [9] simulations where losses are generated using an appropriate Gilbert model are carried out to study the performance of SP and MP streaming with different optimization objective functions. Each simulation emulates the streaming of a

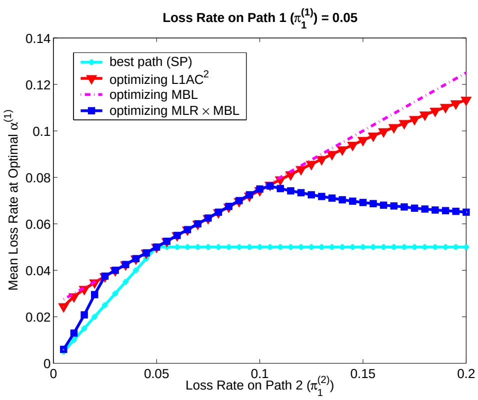


Figure 2: Loss Rate at Optimal

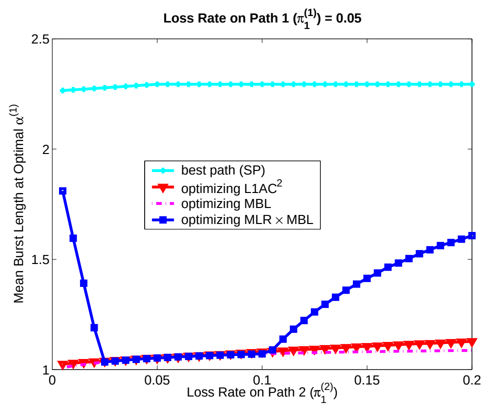


Figure 3: Mean Burst Length at Optimal

2 hours media file which requires a sending rate of 120 pkts/sec. The resulting data sending rate is increased to account for the addition of redundant information, i.e., based on the percentage of FEC overhead. The optimal load distribution α^* is obtained analytically using this adjusted sending rate. FEC is performed on the received packet sequence and the mean information loss rate (MILR), which is the loss rate of *media data* (i.e., not including loss rate of redundant packets) after FEC processing, is measured.

Experiment 3: Optimal Load Distribution and Information Loss with FEC: To compare the effects of different objective functions on α^* when FEC is used, we compute the optimal traffic load distribution using MILR as our optimization objective by a *brute force* search on simulation results. Figure 4 shows the optimal α_1^* as a function of loss rate on Path 2. Figure 5 shows the corresponding MILR. The same experiment is repeated with Path 1 loss rate set to 10%; Figures 6 and 7 show the corresponding α_1^* and MILR, respectively. Each experiment is repeated 10 times with different random seeds, and the results are reported with $95\% \pm 10\%$ confidence intervals. From these results, we make the following observations:

- When Path 1 loss rate is low, i.e., 5%, optimal load distribution on infomation loss increases relatively smoothly; it eventually flattens out as it reaches SP setting.
- When Path 1 loss rate is low, optimizing on MLR*MBL results in a load distribution closer to the optimal based on MILR.
- When Path 1 loss rate is high, i.e., 10%, the MILR-based optimal has a sharper increase when loss rate of Path 2 is close to that of Path 1, and is closer to the load distribution given by SP.
- If a reasonable amount of FEC redundancy is used, optimizing on MLR*MBL gives good performance, i.e., lower information loss rate, and an MLR*MBL system operates closer to the MILR-based optimum. If the path loss rate is high and the amount of FEC redundancy is insufficient, using SP may result in better performance.

Experiment 4: Overhead Requirements for Fixed Information Loss Rate: Here we compare the effect of using FEC on SP (*case i*) and MP streaming, where we use a simple Round-Robin approach (*case ii*) and the MLR*MLR-based optimization approach (*case iii*). Path 1 has a fixed loss rate of 5%. Three scenarios where Path 2 has 0.5%, 5%, 10% loss rates are studied. For SP streaming, we choose the better path, i.e., with a lower loss rate. The objective of this experiment is to find the minimum FEC overhead $(n-k)/k$ required to achieve a resulting quality of service, which in this case we define as a mean information loss rate of $\leq 0.1\%$. (We have tried other values, and the results are qualitatively similar.) For each scenario, different FEC group sizes are tested by varying k. Table 2 gives the mean FEC overhead requirements, averaged over

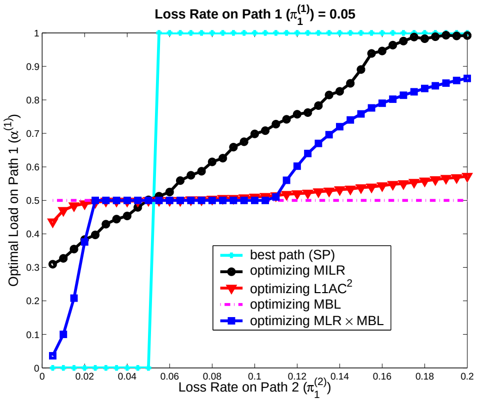


Figure 4: Optimal Load for Different Objective Functions (with FEC)

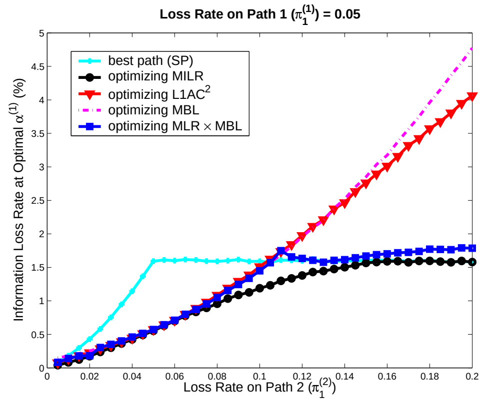


Figure 5: Information Loss Rate at Optimal

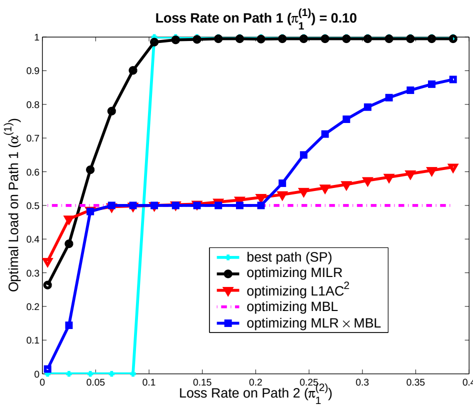


Figure 6: Optimal Load for Different Objective Functions (with FEC, Path 1 loss rate = 10%)

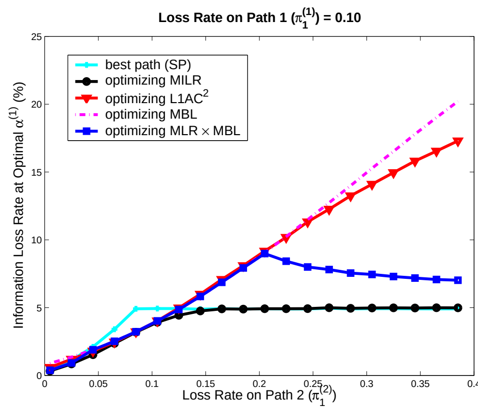


Figure 7: Information Loss Rate at Optimal (Path 1 loss rate = 10%)

10 experiments; the results are reported with 95% ± 11% confidence intervals. From these results, we make the following observations:

- SP, RR, and MLR*MBL have different "best" performing ranges.
- Not surprisingly, RR performs best when paths are homogeneous.
- Performance of MLR*MBL is similar to that of RR for most cases in this experiment.

Experiment 5: Sensitivity Here we study the sensitivity of different optimization objective functions to inaccurate measurements (which are likely to occur in an Internet-type environment), in order to understand the robustness characteristics of the proposed technique. Since our optimization approach is a function of the paths loss rates, we consider cases where the loss rate information is inaccurate. And, specifically, we focus on a comparison between SP and the MLR*MBL-based approach, as above experiments indicate that MLR*MBL is the better of the remaining objective functions. We present results of two cases (1) where Path 2 loss rate measurements are 5% greater than the real path loss rate and (2) where Path 2 loss rate measurements are 5% smaller than the real path loss rate. Figures 8 and 9 depict the resulting MILR loss rate using single path and MLR*MBL, respectively. In each figure we report MILR curves corresponding to

Table 2: FEC overhead needed to achieve information loss rate of $\leq 0.1\%$; Path 1 loss rate = 5%.

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%

the accurate measurements as well as to the two cases of inaccurate measurements. The results are reported with 95% $\pm$ 6% confidence intervals. From these results, we make the following observations:

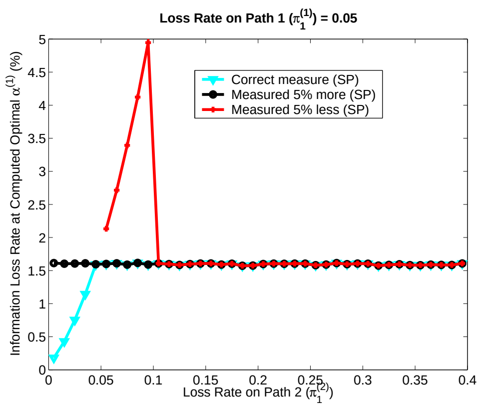


Figure 8: Sensitivity (Path 1 loss rate = 5%, Best Single Path)

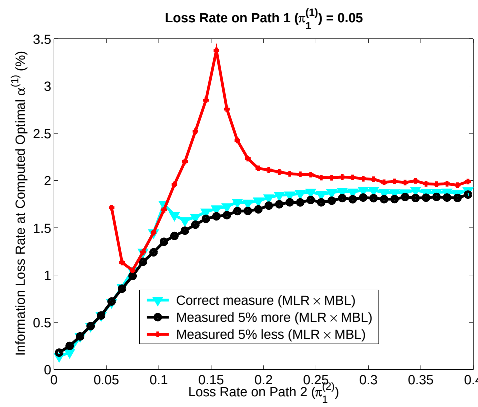


Figure 9: Sensitivity (Path 1 loss rate = 5%, Optimizing MLR*MBL)

- When the loss rate is over-estimated, SP chooses the "wrong" path when the real Path 2 loss rate is low; in this case it is less robust than the MLR*MBL-based approach.
- When the loss rate is under-estimated, both SP and MLR*MBL suffer from inaccurate measurements, although the measurement inaccuracies have a somewhat smaller effect on the MLR*MBL-based approach.

We have also performed experiments where Path 2 loss rates are inaccurate by $\pm$10% and $\pm$20%; those results are qualitatively similar.

Summary: Overall, we observe from the above results that the MLR*MBL-based approach, which attempts a compromise between mean loss rate and loss correlations (which effect the utility of erasure codes), performs well with respect to information loss rate when FEC is used. However, we also observed that the MILR-based optimum load distribution, obtained through brute force searching, shows a smoother transition in load distribution between two paths as compared to the MLR*MBL-based approach. Hence, we believe our approach can be improved in the future. Another observation is that SP performance can be improved with larger FEC group sizes, i.e., values of k, which is not surprising. That is, instead of achieving spatial diversity from multiple paths, one can attempt temporal diversity with the downside of additional delays. However, such additional delays may not suitable for some classes applications which require tighter real-time constraints. A comparison between the delay characteristics of MP streaming and SP streaming with larger FEC group sizes as well as with packet interleaving techniques (which could also achieve temporal diversity) is a topic of future research. In the above experiments, MP streaming is performed with two paths only. These experiments illustrate a noticeable enhancement over SP streaming. Further experiments with a greater number of paths are an ongoing effort. As in [7], we expect further improvement with more paths but with diminishing returns. Future work also includes consideration of importance of different packets as well as addition of path bandwidth constraints to our model.

4 Related Work

We now give a brief survey of existing work on this topic; specifically, we mostly focus on those that consider loss characteristics of continuous media and can be deployed over best-effort networks (as these are considerations in our work as well). Other works, such as [5, 17], consider different applications of multi-path transmission, such as simultaneous downloads and network-level issues.

Earlier efforts on dealing with losses through the use of multiple independent paths include dispersity routing, e.g., [12], where the focus was on reducing delay in file transfers. An important difference in our work is that we focus on streaming applications where the data transmission rate is determined by the application's needs rather than on delivering the data to its destination as fast as possible. Hence, in our case the data is sent through the network at a specific rate and that has an effect on loss characteristics, which we investigate here.

Another set of works considers higher level mechanisms but requires assistance from the lower layers and/or assumes significant knowledge of network topology and/or link capacities and delays (on all links used for data delivery). Given such knowledge, algorithms are proposed for selecting paths which can avoid congested routes, e.g., in [4, 3, 19]. In contrast, our approach does not rely on specific knowledge of topologies, capacities, delays, etc., and only considers whether a set of paths do or do not share joint points of congestion, as can be detected at the end-hosts, e.g., by using the techniques in [18].

Recent literature on this topic also includes works on voice-over-IP type applications. For instance, [11, 10] proposes a scheme for real-time audio transmission using multiple independent paths between a single sender and a single receiver, where multiple description coding (MDC) is used in multi-path delivery and a FEC approach is used in single-path delivery. In the case of using multiple paths, the authors use two equal-sized descriptions of the audio, sending one on each path. In contrast, we believe that it is important to understand the effects of multi-path delivery on loss characteristics, even without the use of coding techniques (an advantage being that it can then be made to work well with many coding techniques or media codecs). We also focus on cases where traffic is not necessarily distributed equally on separate paths, i.e., on the load distribution problem.

We also note that "live" applications (such as voice-over-IP) have different characteristics than pre-recorded applications (as we are considering here). For instance, one such difference is the need to disperse data in real-time, whereas in our case, we can distribute it to the multiple senders ahead of time. Another difference might be the ability to address the potentially adverse effects of MP streaming on delay characteristics (as mentioned in Section 1).

Another work [1] uses multiple description coding in addition to multiple paths, as opposed to using FEC, thereby tying the benefits of the technique to a particular coding scheme. In contrast, as already stated we believe that it is important to understand the benefits of MP streaming independently of data recovery schemes used. The applicability of this study (at least to future streaming applications and network conditions) is also somewhat limited in that (a) it constraints the bursts on paths to no more than 3 and (b) it only considers cases where the loss rates across multiple paths are fixed at 5%.

To the best of our knowledge, the work in [7] is the first to provide an analytic framework for illustrating the benefits of MP streaming as compared to SP streaming, using metrics which describe both mean loss rates as well as correlations between losses (as illustrated in our study as well, it is important to consider both). In [15] the authors also consider delivery of pre-recorded video from multiple senders distributed across the network. However, this work focuses on a TCP-friendly optimization algorithm for (a) rate distribution among the paths (i.e., how much data to send over each path) and (b) packet distribution among the paths (i.e., which packet should be sent over which path), with the objective of minimizing the loss rate at the receiver and the added delay due to the distribution of the media source, respectively. In [14] FEC techniques are added (as compared to [15]). Again, distribution algorithms are considered but with the objective of minimizing the probability of irrecoverable error. In contrast, due to the nature of the application, we believe that in investigating when multi-path streaming is beneficial, it is also important to consider loss characteristics even when the losses cannot be fully recovered because, e.g., video can be displayed under some losses. In [16], the authors use relay nodes to provide multiple paths for delivery of streaming media (this is also suggested as a possibility in [7]). To ensure diverse paths, their system requires detailed information about the underlying network topology, e.g., router names and link properties. In contrast, we do not require such information.

5 Conclusions

In this work, we considered the load distribution problem in the context of multi-path streaming. Specifically, we focused on appropriate optimization objectives for splitting the continuous media traffic among multiple paths, based on path characteristics, where the high level goal is to improve the quality of the received media. We proposed several optimization objectives and evaluated their characteristics through analysis and simulations. Our results indicate that approaches which minimize the mean packet loss rate may increase the loss correlation, and vice versa. These results also indicate that both frequency of losses and their correlations are important to the resulting quality of streamed media, when erasure codes are used. We also found that optimizing on MLR*MBR provides a reasonable compromise between these two factors. Moreover, our results illustrate that significantly higher levels of redundant information are needed under single best path streaming in order to achieve the same level of streaming quality as multi-path streaming (i.e., without incurring additional delays). The robustness of the optimization schemes, when path characterises are inaccurately measured is studied as well. We believe that these results are useful in providing guidelines for multi-path streaming systems design, and they serve as an important step for future multi-path streaming research directions.

References

[1] J. Apostolopoulos, T. Wong, S. Wee, and D. Tan. On multiple description streaming with content delivery networks. In *The IEEE INFOCOM*, New York, New York, June, 2002.

[2] J.-C. Bolot, S. Fosse-Parisis, and D. Towsley. Adaptive FEC-Based Error Control for Internet Telephony. In *INFOCOM*, 1999.

[3] J.-C. Chen and S.-H. Chan. Multipath routing for video unicast over bandwidth-limited networks. In *the IEEE Globecom*, San Antonio, Texas, November 2001.

[4] H. Chu and K. Nahrstedt. Dynamic multi-path communication for video traffic. In *the Hawiian International Conference on System Science*, Hawaii, January, 1997.

[5] I. Cidon, R. Rom, and Y. Shavitt. Analysis of multi-path routing. *IEEE/ACM Transactions on Networking*, 7(6):885–896, 1999.

[6] D. L. Gall. MPEG: a Video Compression Standard for Multimedia Applications. *Communications of the ACM*, April 1991.

[7] 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.

[8] K. Harfoush, A. Bestravos, 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.

[9] http://www.mesquite.com/. *CSIM18*.

[10] Y. J. Liang, E. G. Steinbach, and B. Girod. Multi-stream voice over ip using packet path diversity. In *the IEEE Fourth Workshop on Multimedia Signal Processing*, Cannes, France, October 2001.

[11] Y. J. Liang, E. G. Steinbach, and B. Girod. Real-time voice communication over the internet using packet path diversity. In *the ACM Multimedia Conference*, Ottawa, Canada, September/October 2001.

[12] N. F. Maxemchuk. Dispersity routing. In *the IEEE International Conference on Communications*, San Francisco, California, June 1975.

[13] P. Morse. *Queues, Inventories, and Maintenance*. John Wiley, 1958.

[14] T. Nguyen and A. Zakhor. Distributed video streaming with forward error correction. In *the International Packetvideo Workshop*, Pittsburg, Pennsylvania, April 2002.

[15] T. Nguyen and A. Zakhor. Distributed video streaming over internet. In *the SPIE Conference on Multimedia Computing and Networking*, San Jose, California, January 2002.

[16] T. Nguyen and A. Zakhor. Path diversity with forward error correction (pdf) system for delay sensitive applications over the internet. In *The IEEE INFOCOM*, San Francisco, California, March, 2003.

[17] P. Rodriguez, A. Kirpal, and E. Biersack. Parallel-access for mirror sites in the internet. In *INFOCOM*, Tel Aviv, Israel, March, 2000.

[18] D. Rubenstein, J. Kurose, and D. Towsley. Detecting shared congestion of flows via end-to-end measurement. In *the ACM Sigmetrics Conference*, Santa Clara, California, June, 2002.

[19] H. Tahilramani, S. Kalyanaraman, A. Weiss, S. Kanwar, and A. Ghandi. Bananas: An evolutonary framework for explicit and multipath routing in the internet. In *the ACM SIGCOMM FDNA*, Karlsruhe, Germany, August, 2003.

APPENDIX A

Experiment (Effect of Correlated Bursty Losses on Video Quality): In this experiment, we drop 2% of the frames from video $\mathcal{V}$. These 2% losses are introduced in a variety of "patterns", e.g., the dropped frames can be evenly spaced throughout video $\mathcal{V}$ or they can be more bursty. The details of which frames are dropped, given a particular drop pattern as identified by the burst length, are given in the first two columns of Table 3. Moreover, in evaluating the quality of the resulting video $\mathcal{V}$, we use a common error concealment scheme to make up for a dropped frame. Specifically, a dropped frame is replaced by the previous frame which is successfully received. For example, frame i replaces frames $i+1, i+2, \cdots, i+k$ if frame i is received successfully and frames $i+1, \cdots, i+k$ are loss.

For each possible frame loss pattern, we measure the quality of the received video by computing the corresponding peak signal-to-noise ratio (PSNR). (Note that, a larger value of PSNR implies a higher quality of the video.) In general, for a video of l frames where each frame consists of $m \times n$ pixels, (each containing an RGB value[3] with each of the three colors represented by 8-bits), the PSNR is calculated using the following expression (in dB):

$$SNR_{peak} = 10 \times \log_{10} \frac{255^2}{\left(\frac{\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{l} \sum_{c=1}^{3} (P_1(i,j,k,c) - P_2(i,j,k,c))^2}{3 \times m \times n \times l} \right)}.$$

where $P_s(i,j,k,c)$ is the pixel value at coordinate (i,j) of k-th video frame (of stream s, $s = 1, 2$) and color

[3] Information about the three colors, red, green, and blue.

channel c where $c = 1, 2, 3$, for red, green, and blue, respectively. In our experiment, the values of m,n, and l are 352, 240 and 1500, respectively. The source video in this experiment is using MPEG-1 NTSC settings [6] where each frame is 352×240 (with 29.97 frames per second), hence the values of m and n above. Also, we use approximately the first 50 seconds of this video for this experiment, hence the value of l above. Values for P_1 are obtained from the frame sequence resulting after the drop-and-conceal process while values for P_2 are obtained from the original video frames of $\mathcal{V}$. Table 3 gives the PSNR values for the different burst patterns. We can observe that given the same amount of non-recoverable information loss (e.g., 2% in our experiment), the PSNR metric can be significantly lower for the more bursty loss patterns, and hence can the quality of the video. Thus, we believe that burst length distribution and correlations between losses are the right metrics for evaluating the goodness of a streaming approach as they directly reflect on the quality of received video.

Error Burst Length	Lost Frames Numbers	PSNR (dB)
1	25+k*50 where k$\in \{0, 1, \cdots, 29\}$	39.107 dB
2	{50,51} + k*100 where k$\in \{0, 1, \cdots, 14\}$	38.015 dB
3	{74,75,76} +k*150 where k$\in \{0, 1, \cdots, 9\}$	31.325 dB
5	{123,124,...,127} +k*200 where k$\in \{0, 1, \cdots, 5\}$	30.433 dB
15	{368,369,...,381,382} +k*750 where k$\in \{0, 1\}$	28.407 dB
30	{736,737,...,764,765}	29.942 dB

Table 3: Peak signal-to-noise ratio (PSNR) for various bursty loss patterns with a constant (2%) loss rate.

APPENDIX B

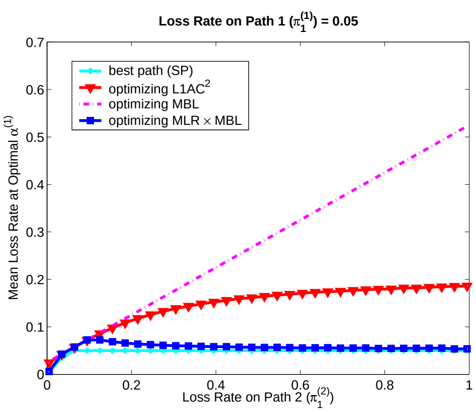


Figure 10: Loss Rate at Optimal

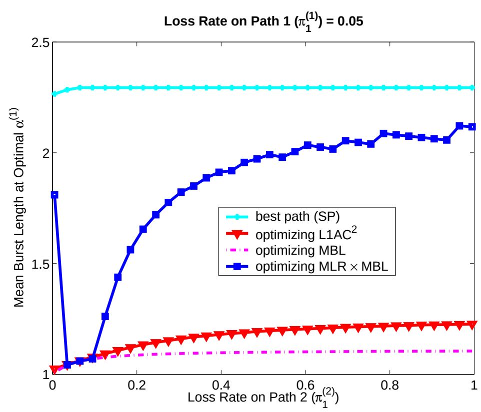


Figure 11: Mean Burst Length at Optimal

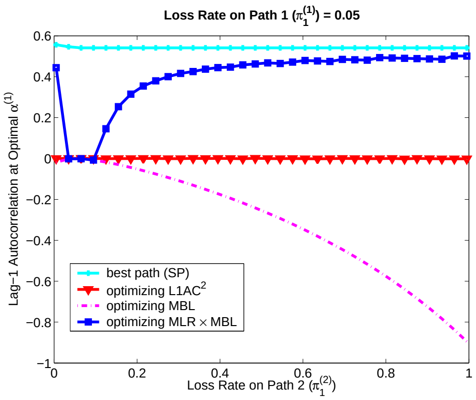


Figure 12: Lag 1 at Optimal

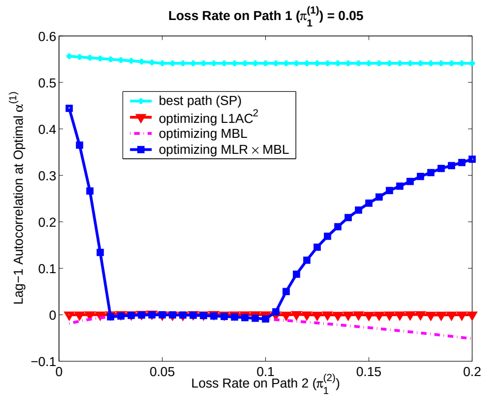


Figure 13: Lag 1 at Optimal

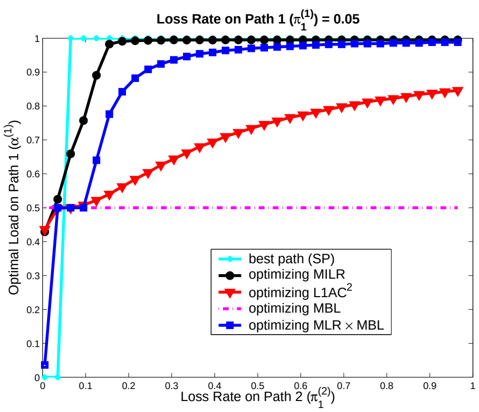


Figure 14: Optimal Load for Different Objective Functions (with FEC)

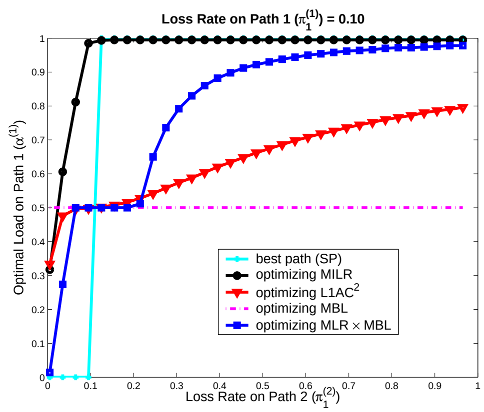


Figure 15: Optimal Load for Different Objective Functions (with FEC, Path 1 loss rate = 10%)

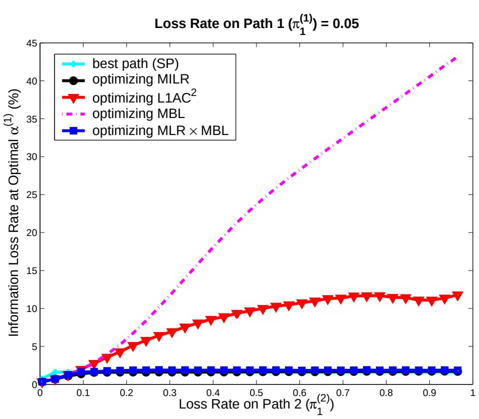


Figure 16: Information Loss Rate at Optimal

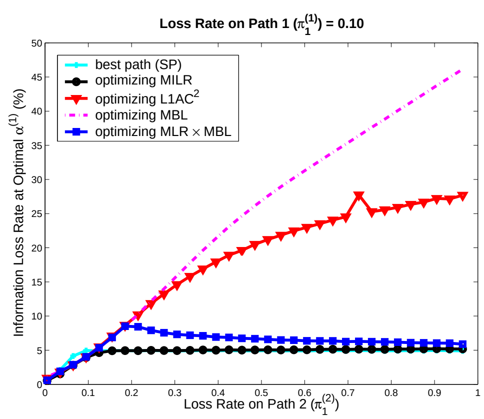


Figure 17: Information Loss Rate at Optimal (Path 1 loss rate = 10%)