TEL-AVIV UNIVERSITY
RAYMOND AND BEVERLY SACKLER
FACULTY OF EXACT SCIENCES
SCHOOL OF MATHEMATICAL SCIENCES

Packet Dispersion and the Quality of Voice over IP Applications in IP networks

Thesis submitted in partial fulfillment of the requirements for the M.Sc. degree in the Department of Computer Science, Tel-Aviv University

By

Zlatokrilov Haim

The research work for this thesis has been carried out at Tel-Aviv University under the supervision of Prof. Hanoch Levy

May 2003

Contents

List of Figures

Glossary

g_{min}	Burst size in E-model
I_e	Loss impairment in E-model
I_d	Delay impairment in E-model
$NL_i^{(\delta)}(k)$	δ Noticeable Loss indicator, for packet k of session i.
δ	Noticeable loss constraint distance
P	Number of paths
N	Number of sessions
$NLR_i^{(\delta)}(K)$	*NLR* for session *i* with loss constraint δ, and for a sequence of *K* packets
$\overline{NLR}^{(\delta)}$	Average *NLR* for N sessions over the pates
$l(t)$	Loss indicator function on packet *t*
L_i	Loss probability on path p_i
$\overline{L}_i$	Success probability on path p_i
$Q(k)$	A function that defines path index of packet *k* in a *periodic dispersion* strategy.
$c_{i,j}$	Fraction of packets belonging to session s_i that are sent on path p_j
$\rho_{i,j}$	Probability that packets of session s_i are sent on path p_j
$\widehat{L}_i$	Total loss experienced by session s_i in *random dispersion* $\widehat{L}_i = \sum_{k=1}^{P} \rho_{i,k} L_k$
$S_i(t)$	State (in Gilbert model) on time *t* on path p_i
P_{G_i}	Loss probability is state *G*, on path p_i
P_{B_i}	Loss probability is state *B*, on path p_i
A_i	State transition matrix for path p_i
B_i^1	Vector representing the loss probability conditioned on the path state
$\pi_i^T(t)$	Transpose of the state probability vector at time *t*
π_i	Steady state probability vector of the Markov chain
α_i	Transition probability from state *G* to state *B*, on path p_i

β_i	Transition probability from state *B* to state *G*, on path p_i
ϕ	Denotes either loss or success ("don't care")
$L_i(t)$	Random variable denoting the event of loss or success at time *t* on path p_i
$l_i(t)$	Actual event occurring at *t* on p_i
$E_i(t,n)$	An event sequence starting at *t* on length *n*, on p_i
$\widehat{\mathrm{A}}_i^{l_i(t+j)}$	Conditional transition matrix in $l_i(t+j)$, on p_i
T_i^Q	Transition matrix for periodic dispersion policy *Q*
p'	An "equivalent path" in *random dispersion* under Gilbert loss model
S'	Combined state in *random dispersion* under Gilbert loss model
Γ	Transition matrix for p'
$\hat{\pi}$	Steady state probability vector of p'
$\hat{B}^1$	Diagonal matrix of loss probability on p'
T_B	Average duration time for the chain to be in states *B*
T_G	Average duration time for the chain to be in states *G*

Abstract

Next Generation Networks (NGN) and the migration toward IP networks is likely to make the IP technology the main vehicle for carrying voice and video calls on modern networks. Examining the quality of Voice and Video over IP (V^2oIP) is critical for the successful deployment of these technologies. Packet dispersion is a mechanism by which the packets of a certain session are dispersed over multiple paths, in contrast to the traditional approach by which they follow a single path most of the time. In this work we examine the quality of a Voice over IP (VoIP) application and the effects of packet dispersion on it. We focus on the affect of the network loss on the application, where we propose to use *Noticeable Loss Rate* (*NLR*) as a measure correlated with the voice quality. We analyze the *NLR* for various packet dispersion strategies over paths experiencing Bernoulli (memory less) or Gilbert (bursty) loss models, and compare them to each other. Our findings show, that in many situations the use of packet dispersion *reduce*s the *NLR* and thus improves session quality. The results suggest that the use of packet dispersion may be useful for these applications.

1 Introduction

Next Generation Networks and today's migration towards IP based networks is likely to make these networks the main infrastructure for carrying Voice and Video applications. A major issue that needs to be solved to make this migration successful is that of the required quality of the applications over the "best effort" IP network. In this study we examine the affect of packet dispersion over multi-path on VoIP application. Packet dispersion[1] in IP networks, is a mechanism in which application packets are dispersed between parallel paths leading from the source to the destination, based on a predefined dispersion strategy. The dispersion can be implemented by the source application (e.g. by using source routing or other techniques) or by nodes in the network. For example, multi-homing devices and content delivery network (CDN) companies, such as Akamai, that use edge architecture to achieve load-balancing and improved network utilization.

Traffic dispersion techniques are used in many technologies, for a variety of reasons. In CDMA radio channels, traffic dispersion (also called frequency-hopping) is used for security reasons and in order to statistically multiplex noises, so that if a few out of *n* channels are blocked or noisy, the errors on theses channels will be spread to all users, making better resource utilization. The same idea of traffic dispersion in IP networks is suggested to reduce traffic burstiness and therefore achieve higher resource utilization [9][33]. Another idea, proposed in [17], suggests using traffic dispersion as a better method to Forward Error Correction (FEC) technique for voice over IP applications. Traffic dispersion is also used de-facto in IP networks for load balancing purposes.

There are many factors affecting the quality of Voice over IP (VoIP) applications. We can divide these factors into two groups, the technology built-in mechanisms and the underlying network behavior. The technology built-in mechanisms include issues such as codec type and its capabilities, Packet Loss Concealment (*PLC*) mechanisms, Forward Error Correction (*FEC*) etc. The network behavior is usually measured in three measures, packet loss rate, delay and jitter (delay variance). Clearly, as these parameter grow, quality degrades. However, the acceptable

[1] Since we focus our discussion on IP networks, we will use the term Packet Dispersion instead of the general term Traffic Dispersion.

delay, for bi-directional real-time streaming applications, is usually limited by values of 200-250 milli-seconds. Thus, both delay and jitter can be roughly translated, physically and mathematically, into a loss measure, as late packets arriving at the destination are not useable and can be counted as lost. We will concentrate in this study on the packet loss experienced by a session, regardless of the cause of the loss (whether a real network loss or a dropped packet due to late arrival).

Perceptual studies of applications such as IP phones have shown that user dissatisfaction rises dramatically in presence of bursty losses. Average packet loss rate property, as shown in many studies, is not enough to capture the affect of network behavior on VoIP applications. For better quality evaluation one should also take into account loss burstiness and recency effects. Taking these together with the technology built-in mechanisms can lead to a good estimation of VoIP application quality, as suggested in the E-model [6][7][10]. One intrinsic property shown in these studies is that bursty losses degrade voice quality. When thinking of *PLC* and *FEC* mechanisms, this is quite intuitive. Both *PLC* and *FEC* are designed to compensate for losses by using information from the session's history (*PLC* is using modulation on previously arriving voice frames to replace a lost frame and *FEC* uses data from previous packets), bursty losses clearly harm the effectiveness of these mechanisms. Intuitively, our ears behave similarly, we may not notice an isolated noise but a relatively long noise will disturb the conversation. Due to these properties we conclude that in many situations the packet *Loss-Rate* measure should be replaced by the *Noticeable-Loss-Rate (NLR)* measure [18] as the basic ingredient in computing the perceived quality of VoIP applications. The *NLR* metric counts losses of 'close' packets and ignores losses of distant packets. However, we do not claim that the *NLR* is a sufficient parameter for capturing network behavior for the evaluation of quality, but that based on [6][7][10][32], the *NLR* is a metric well correlated with perceived voice quality (the lower the *NLR* the better the quality). Therefore, in this work we focus on the *NLR* experienced by VoIP sessions.

Taking all this together, it seems that some of the advantages of traffic dispersion mentioned above can be effective for VoIP applications. The fact that packet dispersion reduces "noise" (losses in our case) burstiness can potentially improve perceptual quality[2]. We examine this fact by calculating and comparing the *NLR* experienced by sessions delivered by various dispersion strategies and by comparing the results to the traditional delivery strategy (no-dispersion).

The analysis in this work is based on assuming that the loss experienced in the network follows either a Bernoulli (memory less) process or a Gilbert loss model, which expresses loss burstiness. Analyzing various packet dispersion strategies for Bernoulli loss model, we demonstrate that the packet dispersion reduces the *NLR* in most of the practical cases. An examination of the *NLR* under the Gilbert model leads to the conclusion that in many cases packets dispersion can highly reduce *NLR,* but in other cases, depends on path characteristics, there are opposite results. Though the results show that packet dispersion is beneficial in many cases for VoIP, we are aware of the fact that packet dispersion has many side effects and may cause other network problems (e.g. out of order packets), which may harm other applications[3]. Thus, devices implementing packet dispersion should take into consideration the specific application requirements, network conditions over the routes and the dispersion strategies for overall enhanced network performance. It is worthwhile to mention that traffic dispersion can also be used for QoS differentiation and enhanced network utilization purposes over asymmetric paths. We present in this study considerations for network planning using packet dispersion to assure maximal network utilization.

Our analysis provides the mathematical machinery needed for computing the NLR experienced by the sessions in these systems. The results are expressed in expressions whose computational complexity is very small (linear), despite having a very large (exponential) state space.

The structure of this work is as follows: In Section **2** we introduce the Noticeable Loss Rate model adopted from [18] and the underlying assumptions in our model. We then turn into mathematical analysis of packet dispersion strategies under the Bernoulli loss model (Section 3) and under the Gilbert model (section 4). For both loss models, we first analyze the *NLR*

---

[2] The idea of spreading errors in order to improve quality comes in the form of reordering packets to overcome bursts for VoIP, as presented in [8].

[3] In [2] it claimed that given the loss rate, the performance of TCP applications improves when losses tend to appear in bursts. Meaning that the same affect of reducing burstiness that is beneficial for VoIP is bad for TCP.

experienced by a session traversing a single path (no-dispersion), as is typically the case in traditional networks. We then turn to analyze the *NLR* as experienced in multi-path environment, and examine two typical packet dispersion scheduling policies: i) The *memory-less random packet scheduling*, in which the paths taken for the packets of a stream are chosen using a memory-less probabilistic mechanism (selection from a predefined set of paths), and ii) The *periodic packet scheduling* in which the paths taken for a packet stream are selected according to some periodic order; a common special case of the latter scheduling is the *Round-Robin* scheduling. Having analyzed these systems we then compare them to each other and bring numerical results to support our findings.

2 Model and Notations

2.1 Voice quality, the factors affecting it and its evaluation

Traditionally, voice perceived quality is measured by the *Mean Opinion Score* (*MOS*) or by mechanical techniques such as PESQ [13] and PSQM [12]. Another non-intrusive monitoring technique for VoIP, incorporating the effects of time varying packets loss and "recency", based on the E-model [9] is suggested in [6][10][32].

There are many factors affecting voice quality in VoIP applications. In general, one can divide these factors into application factors (e.g codec type, jitter buffer implementation, etc.) and network performance factors: delay, jitter and loss. The techniques suggested in [6][32] propose that given the codec type and other application parameters, loss (I_e) and delay (I_d) impairments are the main factors affecting voice quality. From these impairments we can compute the gross score called R value which can be mapped to MOS. The delay impairment causes relatively small affect as long as it is bounded within certain constraints (usually up to 250ms, see [10]). Roughly speaking, this factor can be used to translate network delay into network loss by counting all the packets whose delay exceeds a certain threshold as lost packets. This results with network loss being the major network performance parameter affecting voice quality.

The average packet loss rate metric alone is not enough to determine voice quality. The other factors, reminded in [6][32], are the *recency* effect (the location of the lost frames, e.g. loss at the end of the session significantly degrades perceived quality in comparison to losses at the beginning of the session) and the loss burstiness (a packet is considered to be in a burst if less than g_{min} packets have arrived since the previous packet was lost). The burstiness, having the greater effect, can reduce MOS in more than one grade (out of five) as presented in Figure 1, taken from 0[17] (similar results are shown in [6]):

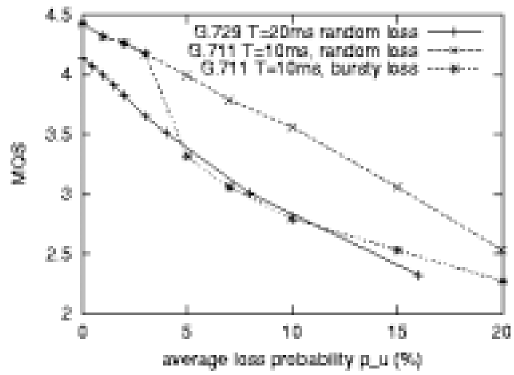


Figure 1: Call Quality (MOS) as a function of average loss probability

Perceptual studies, such as those mentioned in [8], also support the fact that bursty losses may dramatically reduce perceptual quality, especially for audio. In modern codecs internal *Packet Loss Concealment* (*PLC*, see [11]) algorithms are used to reduce the affect of packet loss on perceived quality. When a loss occurs the decoder derives the data of the lost frame from previous frames to conceal losses. A simple example of a *PLC* mechanism would be to use the last (properly arrived) packet to replace a lost packet. Some codec concealment mechanisms may be effective for a single lost packet, but not for consecutive losses or bursts of losses. *Forward Error Correction* (*FEC,* see [28]) mechanisms are also used to compensate for lost packets by appending the information of previous voice frames to packet payload. Clearly, for this technique sequential losses decrease *FEC* efficiency and reduce voice quality.

We thus conclude that the loss rate and loss burstiness are the major network performance factors affecting voice quality and we focus on their performance. Next we define and discuss the *Noticeable Loss Rate (NLR*) as a measure for loss burstiness well correlated with voice perceived quality.

2.1.1 Noticeable Loss Rate (*NLR*)

The *IP Performance Metrics* (ippm) working group in the IETF has proposed a set of metrics for packet loss [19]. This includes loss constraint distance (i.e the threshold for distance between two losses) and the *Noticeable Loss Rate* (*NLR*) metric that is the percentage of lost packets with loss distance smaller than the loss constraint distance. In VoIP applications the loss constraint distance is usually related to the convergence time of the decoder[4]. Clearly, with the growth of *NLR* perceived voice quality decreases.

Definition of *NLR*

The *Loss Distance* is defined (as in [18]) as the difference in sequence numbers between two successively lost packets. The loss event of a packet is defined to be "a δ noticeable loss" event (and is denoted as $NL^{(\delta)}$), if the *loss distance* between the lost packet and the previously lost packet is no greater than δ, where δ, a positive integer parameter, is the *loss constraint* (The *loss constraint* may be equivalent to $g_{\min}$ in [6] determining whether a packet belongs to a *gap* or a *burst*). By choosing δ based on the sensitivity of the *PLC* mechanism, one can measure how 'noticeable' a loss might be for quality purposes (in [6] it is claimed the a typical value for minimal gap size, $g_{\min}$, is 16). For *FEC*, sequential or very 'close' losses will be 'noticeable losses' so that $\delta \le 3$ may be the upper bound for the *loss distance*.

We define the *Noticeable Loss Rate* (*NLR)* as the number of noticeable losses divided by the total number of packets. This definition agrees with the metrics 'Type-P-one-Way-Loss-Distance-Stream' defined in [19], but instead of dividing the number of noticeable losses by the number of losses (or by the number of successfully received packets), we divide it by the total number of packets. This metric seems more suitable for voice applications as the values of lost or received packets are of no importance in this model once counting the noticeable losses. Where necessary we will associate the parameter δ with the notion of noticeable loss rate, reading δ-noticeable loss rate, or $NLR^{(\delta)}$.

The *loss indicator function* for a certain flow reflects the loss event of packet *t:*

$$l(t) = \begin{cases} 1 & \text{if packet } t \text{ is lost} \\ 0 & \text{Otherwise} \end{cases} \quad \textbf{(1)}$$

[4] Note that the *Consecutive Loss Factor* (CLF), mentioned in [8], is a special case of the *NLR* metric.

The event that packet k in session i is a noticeable loss with loss constraint δ, is denoted by indicator function $NL_i^{(\delta)}(k)$:

$$NL_i^{(\delta)}(k)=\begin{cases}1 & l(k)=1 \text{ and } \exists t\in[\mathrm{k}-\delta,\mathrm{k}-1] \text{ where } l(t)=1\\ 0 & \text{otherwise}\end{cases} \quad \textbf{(2)}$$

The noticeable loss rate for session i with loss constraint δ, and for a sequence of K packets is then given by:

$$NLR_i^{(\delta)}(K)=\frac{1}{K}\sum_{s=1}^{K}NL_i^{(\delta)}(s) \quad \textbf{(3)}$$

Next we propose an alternative definition to that given in Eq. **(2)** for the noticeable loss event ($NL_i^{(\delta)}(k)$):

$$NL_i^{(\delta)}(k)=\begin{cases}1 & l(k)=1 \text{ and } \exists t\in[\mathrm{k}+\delta,\mathrm{k}+1] \text{ where } l(t)=1\\ 0 & \text{otherwise}\end{cases} \quad \textbf{(4)}$$

Proposition 1: For any sequence of loss events, the number of noticeable loss events under the definitions **(2)** and **(4)** are identical to each other.

The proposition is proven by counting, under both definitions, the number of losses that are not noticeable and subtract them from the total number of losses.

Due to the proposition, the definition of the noticeable loss rate (Eq. **(3)**) holds the same regardless of the definition of $NL_i^{(\delta)}(k)$.

In the analysis we analyze the system under the assumption of steady state. Thus we have:

$$NLR_i^{(\delta)}=\lim_{N\to\infty}\frac{1}{N}\sum_{k=1}^{N}NL_i^{(\delta)}(k)=\frac{1}{N}\sum_{k=1}^{N}\Pr[NL_i^{(\delta)}(k)=1] \quad \textbf{(5)}$$

In order to conduct a meaningful comparison in a scenarios where multiple sessions are involved, we will evaluate the average NLR taken over the N sessions, denoted $\overline{NLR}^{(\delta)}$ ($\overline{NLR}^{(\delta)}=\frac{1}{N}\left(\sum_{i=1}^{N}NLR_i^{(\delta)}\right)$)

2.2 Modeling loss

Losses at the application level are caused both by the IP network losses and by network delays. In this study, we model the application loss, regardless the source of the loss (network loss or network delay[5]). Internet loss models have been studied in many studies, such as [4][5]. Here we are focusing on modeling the losses experienced by VoIP applications. For that matter we look at these applications as a constant bit rate applications. We assume that time is divided into time slices[6]. In each time slice *t*, a packet is sent by the application. For clarity, in the analysis we refer to the packet sent in time slice *t* as packet *t*. Thus the loss model, expresses the loss experienced by the application.

We will focus on Bernoulli (memory-less) loss model (see section 3) and Gilbert (bursty) loss model[7] (see section 4). The Gilbert loss model is used in many studies to model the bursty loss behavior in the Internet. This bursty loss behavior has been shown to arise from the drop-tail queuing disciplines implemented in many Internet routers.

2.3 Independent Multiple-Paths over packet switched networks

There are several fundamental requirements for implementing packets dispersion. The first is the physical existence of multiple parallel paths between the two endpoints participating in the session, and the second is the ability to disperse application packets among the parallel paths.

The construction of parallel paths can be achieved by using parallel paths in MPLS networks, using Source Routing, constructing static parallel routes in the IP network or any other way, as discussed in [1] and [31][8]. Moreover parallel paths exist de-facto in today's networks via the multi-homing connectivity approach, where load-balancing devices disperse traffic to parallel routes.

We will assume that the losses on the different paths are independent of each other. This is likely to occur if the paths are fully disjoint or if at least the "noisy", in terms of loss and delay,

---

[5] Roughly, we may say the packets delayed beyond 250ms are considered lost.
[6] Usually in duration of 10 to 30 milli-seconds in VoIP applciaitons.
[7] Other known loss models such as Extended Gilbert Model, general Markov chain model, heavy-Tailed distribution of packet loss and other models described in [16][23], are not used in the analysis and therefore irrelevant for this study.
[8] The construction of independent parallel paths might be problematic in the Internet, but feasible in managed networks.

components of the different paths are disjoint. Theoretically speaking, this assumption can hold in a multi-homing environment in the Internet as well. Packets in the Internet usually cross only few managed networks on the way to destination. Hence, it might be enough for the first domain to disperse the packets between two different managed networks to create the effect of dispersion between independent parallel paths.

We show in this study that packet dispersion can improve VoIP application quality, regardless of how it is realized, whether by a multi-homing device located in the network or by a dedicated dispersing element intended to improve quality. Since packet dispersion can improve quality, it might be worthwhile to place dispersing devices in the network. Such devices should be located on the path between the sender and the receiver and may take automatic dispersion decisions based on current network conditions or base on a-priori knowledge gathered by network management elements[9].

The destination endpoint, in VoIP applications, must be able to receive and synchronize packets arriving from parallel paths and manage the jitter-buffer optimally in order to reduce delay to minimum and handle out-of-order packets (which may be very common if the paths are not of equal delay)[10]. In our model we assume that parallel paths have small delay differences in comparison to the allowed buffering delay. This assumption is likely to hold for many networks, but may be problematic if some of the networks contain satellite links (while the others not). In applications where large buffering is allowed, such as one-way video or voice streaming, the gap in delay may be unimportant and compensated by increased jitter-buffer. For interactive applications that demand quick response (e.g. phone-call) only small buffering is allowed, up to few tenth of milliseconds, and choosing eligible set of paths is crucial.

Packet dispersion can be used for Quality of Service (QoS) purposes under limited network resources as a traffic engineering utility. By applying proper dispersion strategies one can differentiate between sessions of different QoS requirements. E.g. consider paths p_1 and p_2, each

[9] There are several industrial applications that use measurements as basis for routing decisions. Similar mechanisms can be used for the implementation of traffic dispersion.

[10] The issue of handling multiple streams arriving through multiple routes, is also dealt with, in a different context, in [25]. There the problem of interest is that of downloading parts of a file from several servers via FTP.

with capacity of a single session with loss probabilities $L_1 = 0.05$ and $L_2 = 0.01$. If each stream is carried over one of the routes, one stream suffers the *NLR* caused by loss probability of 5% and the second caused by loss probability of 1%. If *random packet dispersion* was implemented, both sessions would suffer the average loss probability of $\overline{L} = 0.03$. When the acceptable *NLR* for both sessions is caused by $0.03 \le L < 0.05$ both sessions will have sufficient quality under packet dispersion. In the case that one session requires $L \le 0.02$ and the second $L \le 0.04$, both sessions can still benefit, if the proper packet dispersion strategy is used.

2.4 Dispersion strategies

Packet dispersion can be implemented through a variety of strategies, of which we focus the following:

1. *Deterministic scheduling*
   i. *Periodic dispersion* – the source endpoint disperses packets in a periodic schedule manner over the routes repeatedly. For example if the schedule is that. *(i, i, i, j, j)* then in every cycle 3 packets in a row are sent over path p_i, and then the following two packets are sent over path p_j, where this schedule repeats cyclically.
   ii. *Deterministic round robin dispersion* – a special case of *periodic dispersion* where packets are sent in round robin over the paths. For example, if two paths are used, all odd index packets are sent over one path while all even index packets are sent over the other path.
2. *Random packet dispersion* – for each packet, the source endpoint picks randomly one of the paths and sends the packet over it.

The traditional delivery of packets over a single path is referred to as a *no-dispersion* strategy. We will assume that the packet dispersion strategies are executed in session context[11].

[11] This assumption is not mandatory since *random dispersion* or *periodic dispersion* of all packets, regardless of the application, will lead in many cases to the same results.

3 Performance Analysis – NLR under Bernoulli Loss Model

The aim of this section is to evaluate the effect that packet dispersion has on application performance, where the network paths experience Bernoulli (memory-less) losses. To this end we evaluate the *NLR* for sessions traversing a single or multiple paths, for a variety of packet dispersion strategies. We will consider situations, which possibly consist of *N* streams, denoted $s_1 \cdots s_N$, and possibly are routed over *P* parallel paths, denoted $p_1 \cdots p_P$.

In the Bernoulli loss model, each packet *t* shipped over path *i*, has the probability of L_i to be lost. Note that this model is memory-less and thus the loss event of packet *t* is independent of the loss events of other packets. Thus we have:

$$L_i = \Pr[\text{packet } t \text{ is lost on path } i]$$
$$= \Pr[\text{packet } t \text{ is lost on path } i \mid \text{packet } t\text{-}1 \text{ is lost on path } i] \quad \textbf{(6)}$$

3.1 The NLR under No-Dispersion

From Eq. **(4)** the probability of noticeable loss can be given as:

$$\Pr[NL^{(\delta)}(k)=1] = \Pr[l(k)=1] - \Pr[l(k)=1, l(k+1)=0, \ldots l(k+\delta-1)=0, l(k+\delta)=0] \quad \textbf{(7)}$$

As we do the analysis under the Bernoulli (memory-less) loss model: $\Pr[NL^{(\delta)}(k)=x] = \Pr[NL^{(\delta)}(k+t)=x] \ \ \forall t,\ x \in \{0,1\}$. Thus, index *k* can be omitted in Eq.**(5),** and the following representation of $NLR^{(\delta)}$ stands:

$$NLR_i^{(\delta)} = \Pr[NL_i^{(\delta)} = 1] \quad \textbf{(8)}$$

Below we assume that each session is directed over a single path (*no-dispersion* strategy). In the *Bernoulli loss model* the loss probability, L_i, for all packets sent over p_i is given. Based on **(8)**, the *NLR,* when the system is under steady state, experienced by session s_i sent over p_i is:

$$NLR_i^{(\delta)} = \Pr[NL_i^{(\delta)} = 1] = L_i - L_i(1-L_i)^{\delta} \quad \textbf{(9)}$$

Now, assuming that each session takes a different path, that is that without loss of generality session s_i takes path p_i (with loss probability L_i), the expected network *NLR* for the *N* sessions, $\overline{NLR}^{(\delta)}$, is then:

$$\overline{NLR}^{(\delta)} = \frac{1}{N}\sum_{i=1}^{N}\left(L_i(1-(1-L_i)^{\delta})\right) \qquad \textbf{(10)}$$

3.2 The NLR Under Periodic Packet Dispersion

In *periodic dispersion*, packets of session s_i are dispersed over the paths according to a fixed policy. Consider a *periodic dispersion* policy *Q*, with period length *K*. The policy is defined by $Q(k)$ ($Q(k)\in\{1...P\}$ and ($k=1...K$)), meaning that packet *k* in the period will always be sent on $p_{Q(k)}$ periodically. Thus, the path taken for packet *t*, without loss of generality, is $p_{Q((t \bmod K)+1)}$. The *NLR* for session s_i, starting in an arbitrary location of the period is then:

$$NLR_i^{(\delta)} = \frac{1}{l}\sum_{k=1}^{l} L_k\left(1-\prod_{j=1}^{\delta}(1-L_{Q((k+j \bmod K)+1)})\right) \qquad \textbf{(11)}$$

where L_k is the loss probability over the path p_k taken by the session.

For simplicity of analysis consider *periodic dispersion* where the period length is a multiple of $(\delta+1)$. Note that for practical purposes, in many cases one can select the period according to this rule. Given the *periodic dispersion* selected, let $c_{i,j}$ ($\sum_{i}^{N} c_{i,j} = 1 \ \forall i$ and assume $c_{i,j}\delta$ is an integer) denote the fraction of packets belonging to session s_i that are sent on path p_j $j=1...P$. The *NLR* experienced by session s_i is:

$$NLR_i^{(\delta)} = (\sum_{j=1}^{P} c_{i,j}L_j)(1-\prod_{k=1}^{P}(1-L_k)^{c_{i,k}\delta}) \qquad \textbf{(12)}$$

Note that the *NLR* experience by s_i is not affected by session s_j, Therefore the expected *NLR* for *N* sessions over *P* routes is then given by:

$$\overline{NLR}^{\delta} = \frac{1}{N}\sum_{i=1}^{N}\left((\sum_{j=1}^{P} c_{i,j} L_j)(1 - \prod_{k=1}^{P}(1 - L_k)^{c_{i,k}\delta}) \right) \qquad \textbf{(13)}$$

Under limited resources (e.g. the total capacity of paths equals to or approximately equals to the required sessions' payload), *periodic dispersion* can be used for QoS purposes by spreading the sessions in a way that as many sessions as possible will meet their QoS requirements. Finding the optimal periodic dispersion assignment is a problem left for further study.

Note that the *no-dispersion* strategy **(10)** is a special case of the periodic strategy **(13)** where $c_{i,i} = 1 : \forall i$.

The *deterministic round robin dispersion* strategy is a special case of the *periodic dispersion*. We here present a special case[12] where *N*=2 , *P*=2, $c_{i,j} = \frac{1}{2} : i, j = 1,2$ and even δ, that will be used in comparing the strategies. The $\overline{NLR}^{(\delta)}$ is then given by:

$$\overline{NLR}^{(\delta)} = \frac{1}{2}\left\{ (L_1 + L_2)(1 - (1 - L_2)^{\frac{\delta}{2}}(1 - L_1)^{\frac{\delta}{2}}) \right\} \qquad \textbf{(14)}$$

The derivation for odd δ is similar.

[12] E.g. all odd packets are sent over p_1 and even packets over p_2.

3.3 The NLR Under Random Dispersion

In *random dispersion* the decision regarding over which path to send packet *t* of session s_i, is done in a random fashion. Let $\rho_{i,j}$ ($\sum_{j}^{P}\rho_{i,j}=1$) denote the probability that packets of s_i are sent on path p_j. Then the *NLR* experienced by session s_i is given by:

$$NLR_i^{(\delta)}=\sum_{j=1}^{P}\rho_{i,j}L_j\left(1-\left(1-\sum_{k=1}^{P}\rho_{i,k}L_k\right)^{\delta}\right) \quad \textbf{(15)}$$

Under the *random independent* strategy we assume that the path selection of one session is independent of that of another session. Under this setting the loss experienced by the t^{th} packet of s_i is independent of the loss experienced by the t^{th} packet of s_j. Further, the loss of the $(t+1)^{st}$ packet is independent of the loss of the t^{th} packet. The average *NLR* over all sessions is then:

$$\overline{NLR}^{(\delta)}=\frac{1}{N}\sum_{i=1}^{N}\left(\widehat{L}_i(1-(1-\widehat{L}_i)^{\delta}\right), \quad \textbf{(16)}$$

where $\widehat{L}_i=\sum_{k=1}^{P}\rho_{i,k}L_k$.

3.3.1 The *NLR* Under *Random Dispersion* with Limited Resources

Consider the *random dispersion* where the system resources are limited. That is, the combined paths capacity equals, or approximately equals, to the sessions' payload. Thus, the *NLR* of session s_i is dependent on the *NLR* of session s_j through the sharing of the resources.

Consider the case of *N* sessions and *P* paths having together the capacity to carry exactly *N* sessions. For simplicity assume that *P*<*N*. The source endpoint can choose one of $\binom{N}{P}$ possible dispersion combinations for assigning sessions over the paths. The formulation is similar to that given in Eq. **(15)** where: $\sum_{j}^{P}\rho_{i,j}=1$ and $\sum_{i}^{N}\rho_{i,j}$ equals to the number of sessions within the capacity of path p_j. The *NLR* observed by each session depends only on the loss probabilities of the paths it travels over, and is similar to the case of *random dispersion* (see Eq. **(16)**)

Note that the *NLR* of session s_i depends on the *NLR* of session s_j. But this dependency is taken into account in the calculation of $\rho_{i,j}$. Once $\rho_{i,j}$ is set, this model is completely similar to the *NLR* observed in the *random dispersion* model without any path capacity limitations.

To demonstrate how the transmission probabilities can be set, consider two sessions s_1 and s_2, and two parallel paths p_1 and p_2, each with the capacity of one session. There are two possible combinations for sending the packets: 1) Send s_1 over p_1 and s_2 over p_2, and 2) Send s_1 over p_2 and s_2 over p_1. To meet the objective of sending $\rho_{1,1}$ packets of s_1 over p_1 and $1-\rho_{1,1}$ over p_2 (with complement probabilities for s_2), the first dispersion combination should be given probability of $\rho_{1,1}$.

3.4 Comparison of Dispersion Strategies under Bernoulli loss model

The *NLR,* by definition, is always lower than the average loss rate. The difference between the two measures, on a single path, is given by $L_i(1-L_i)^{\delta}$ which monotonically decreases with δ. Below we compare the *NLR* under the various strategies.

3.4.1 Equal Quality paths

Corollary 1: For $L_1 = L_2 = \cdots = L_N$ all dispersion strategies provide the same *NLR*.

This implies that under the Bernoulli loss model, dispersing packets over paths with similar random loss probabilities has no effect on the VoIP quality. From the practical point of view, the use of *no-dispersion* in a multi-path environment is therefore preferable because it reduces the delay variability, the number of out of order events, etc.

3.4.2 Random Dispersion vs. No-Dispersion

The expression of *NLR* under *random dispersion* is identical to that of *NLR* under *no-dispersion* where the loss parameter L_i is replaced by the average loss experienced by session $s_i, \widehat{L}_i$. This means that *random dispersion* in practice averages out the loss over all paths.

One should note that *random dispersion* may or may not be superior to *no-dispersion*, depending on the quality requirements and on the loss parameters. In particular, dispersion may be advantageous since it allows the sessions to share poor quality paths and still stand within quality requirements, while if *no-dispersion* is implemented all sessions traveling over poor quality paths may be of unacceptable quality. The problem for achieving optimal results for *random dispersion* (especially under limited resources as mentioned in 3.3.1) is left for further study.

To demonstrate the tradeoff between these strategies, consider the case of two sessions that need to be delivered over two parallel paths with limited resources (for simplicity consider capacity of single session on each path). Figure 2 and Figure 3 represent the reduction of average *NLR* under *random dispersion* in comparison to *no-dispersion*. Figure 2 shows that for $\delta = 2$ (e.g. for *FEC*) *random dispersion* is better than *no-dispersion* if in both paths loss probability is under 40%, and can reduce *NLR* in up to 13%. Thus for practical reasons (session loss probability is under 20%) *random dispersion* is superior. Figure 3 shows that for $\delta = 16$ (as suggested in [6]) *random dispersion* increases *NLR* if loss probability in both paths is over approximately 43%. For practical purposes *random dispersion* reduces *NLR* in comparison to *no-dispersion* but the gain is relatively low (up to 1.3%). Note that the advantage decreases with the growth in the values of the *loss distances*.

The conclusion is that *random packet dispersion* can be used to improve voice quality.

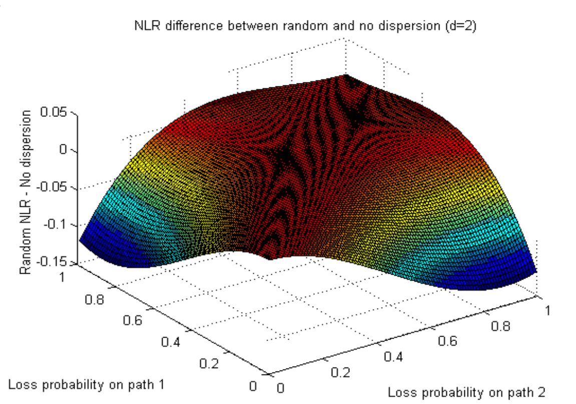


Figure 2: NLR difference between *random dispersion* and no-dispersion for $\delta = 2$

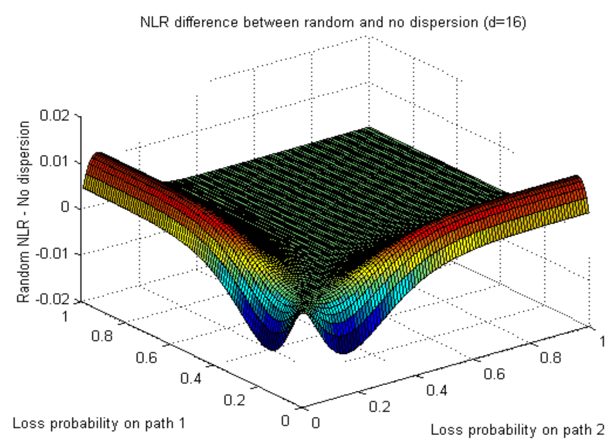


Figure 3: NLR difference between *random dispersion* and no-dispersion for $\delta = 16$

3.4.3 Periodic Dispersion vs. No-Dispersion

There is no simple rule for determining when *periodic dispersion* is superior to *no-dispersion*. It depends on the loss probabilities over the paths and the values of δ. Figure 4 and Figure 5 depict the difference in average *NLR* of two sessions sent over two paths (each with the capacity of a single session).

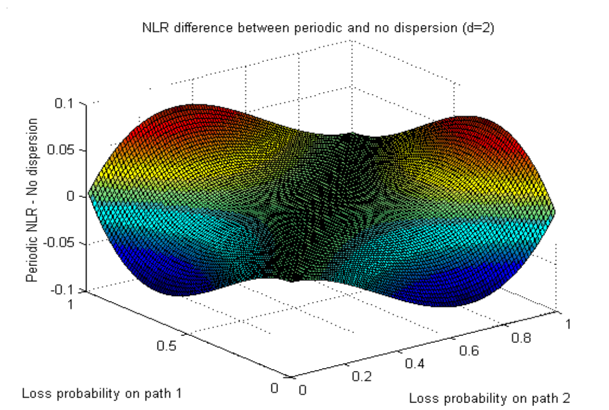


Figure 4: NLR difference between *periodic dispersion* and no-dispersion for $\delta = 2$

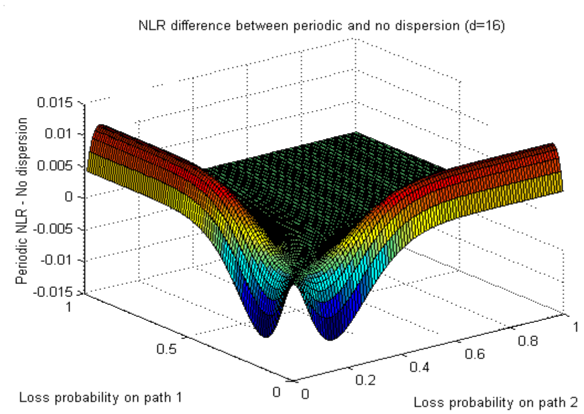


Figure 5: NLR difference between *periodic dispersion* and no-dispersion for $\delta = 2$

According to Figure 4, for $\delta = 2$, *periodic dispersion* can reduce *NLR* if at least one of the paths has loss probability under 50%. Meaning that in the practical range (session loss probability is under 20%), *periodic dispersion* is superior to *no-dispersion* but the gain is small. According to Figure 5, for $\delta = 16$, *periodic dispersion* can reduce *NLR* if both paths have loss probability under approximately 40%. Again in the practical loss range *periodic dispersion* is superior to *no-dispersion*, but the reduction in *NLR* is small.

The conclusion is that *periodic packet dispersion* can improve voice quality in comparison to *no-dispersion*.

Under the same conditions (two sessions to be sent over two paths limited in resources) we present another question: under what values of δ, *deterministic round robin packet dispersion* is better than *no-dispersion*. By comparing **(10)** to **(14),** the values of δ as function of the loss probabilities under which d*eterministic round robin dispersion* is superior *on no-dispersion* given by:

$$\delta < 2\frac{\log(L_1 / L_2)}{\log(1-L_2 / 1-L_1)} \quad \text{for } L_1 \le L_2$$
$$\delta > 2\frac{\log(L_1 / L_2)}{\log(1-L_2 / 1-L_1)} \quad \text{for } L_2 \le L_1 \qquad \textbf{(17)}$$

In Figure 6 the region above the plane represents the values of δ for which *no-dispersion* is superior and the region below the plane represents superiority of *round-robin dispersion*.

Note that for most practical situations, that is, if loss probabilities on both paths are lower than 5%, periodic dispersion is superior for all practical ranges of $1 \le \delta \le 32$. Further, *periodic dispersion* is superior also for loss probability between 5% and 20%, for any $\delta < 8.25$. The Figure also demonstrates (as mentioned in *Corollary* 1) that for equal paths the *NLR* is equal.

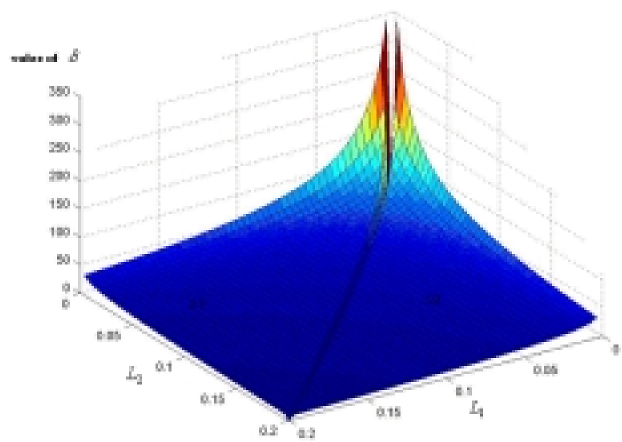


Figure 6: Comparison of *deterministic round robin packet* dispersion and no-dispersion**: Above plane, *no-dispersion* is superior, below plane dispersion is superior**

3.4.4 *Periodic Dispersion* vs. *Random Dispersion*

Corollary 2: *Random dispersion* results in lower *NLR* (and therefore better voice quality) than *periodic dispersion* (where the period length is a multiple of $\delta + 1$) achieved under similar conditions.

Given *periodic dispersion* one can always produce *random dispersion* that results in lower *NLR*. Consider *random dispersion* and *periodic dispersion* where $c_{i,j} = \rho_{i,j}$. This means that the *random dispersion* sends in average the same fraction of packets belonging to session s_i over path p_j. By comparing **(12)** to **(15)**, *random dispersion* results in lower *NLR* since:

$$\prod_{k=1}^{P} \left(\overline{L}_k\right)^{c_{i,k}\delta} < \left(\sum_{k=1}^{P} c_{i,k}\overline{L}_k\right)^{\delta} \quad \textbf{(18)}$$

where $\overline{L}_k = 1 - L_k$. Note that **(18)** holds since the arithmetic weighed average is always greater than the geometric weighted average when $\sum_{j}^{P} c_{i,j} = 1$ (see [27]).

Note that the superiority of *random dispersion* in *NLR* holds despite the fact that the average loss (not *NLR*) is the same for both policies.

Figure 7 and Figure 8 demonstrate the reduction of *NLR* by *random dispersion* in comparison to *periodic dispersion*, when two sessions are sent over two paths (each with capacity of a single session). As can be seen, for the practical loss rates of up to 20%, the reduction in *NLR* is relatively small.

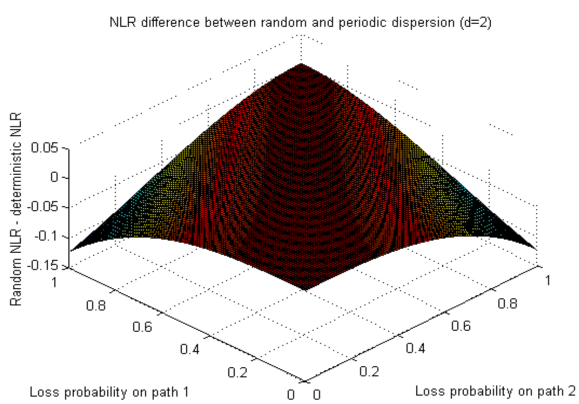


Figure 7: NLR difference between *random dispersion* and *periodic dispersion* for $\delta = 2$

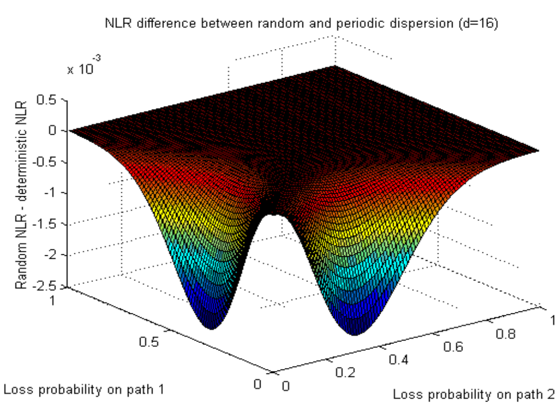


Figure 8: NLR difference between *random dispersion* and *periodic dispersion* for $\delta = 16$

4 Bursty Losses – the NLR under the Gilbert Loss Model

The aim of this section is to evaluate the affect that packet dispersion has on VoIP performance. To this end we evaluate the *NLR* for sessions traversing a single or multiple routes which are subject to bursty losses, for a variety of packet dispersion strategies. Intuitively, packet dispersion can reduce *NLR* and thus improve voice quality, especially over paths suffering bursty losses, since dispersion is expected to spread the losses. We will use the Gilbert loss model to model the bursty losses, over the paths. We will consider a general situation in which *N* streams, denoted $s_1 \cdots s_N$, are possibly routed over *P* parallel paths, denoted $p_1 \cdots p_P$.

4.1 The Gilbert loss Model – A Two States Markov Chain

The loss probability as expressed in the Bernoulli model, is a basic parameter that affects the performance of VoIP applications. However, it is insufficient in capturing loss burstiness which is highly important for these applications. The Gilbert model allows one to express history-dependent losses and thus to capture loss burstiness. This model has been used in many studies to characterize bursty loss in the Internet [4][5][16][17][21].

The model uses a two-state Markov chain to represent the packet losses. We consider a discrete time model where the time unit corresponds to packet transmission for path p_i. Let $S_i(t)$ denote the state of the path at time *t*. We assume that $S_i(t) \in \{G, B\}$ $t = 0, \cdots, \infty$, where *B* stands for Bad and *G* stands Good. The states of the path, $S_i(t)$ are governed by a Markov chain depicted in Figure 9:

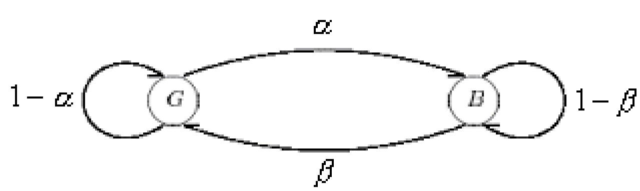


Figure 9: Gilbert Channel model

When the path is in state *G*(*B*) it is subject to Bernoulli loss at rate $P_G\,(P_B)$[13]. Considering path p_i we have:

$$P_{G_i} \overset{\Delta}{=} \Pr[\text{packet } t \text{ is lost over } p_i \mid S_i(t) = G]\,;\; P_{B_i} \overset{\Delta}{=} \Pr[\text{packet } t \text{ is lost over } p_i \mid S_i(t) = B] \quad \textbf{(19)}$$

Clearly $P_{G_i} < P_{B_i}$.

To put this in matrix notation let state 1 represent *G* and state 2 represent *B,* and *let* A_i be the state transition matrix for path p_i, that is $A_i(m,n) = \Pr[S_i(t) = n \mid S_i(t-1) = m]$. Then we have: $A_i = \begin{bmatrix} 1-\alpha_i & \alpha_i \\ \beta_i & 1-\beta_i \end{bmatrix}$.

Let B_i^1 be a vector representing the loss probability conditioned on the path state, that is $B_i^1 = \begin{bmatrix} P_{G\,i} \\ P_{B_i} \end{bmatrix}$. Also let $\mathbf{1} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ and $B_i^0 = \mathbf{1} - B_i^1$.

Note that the Bernoulli loss model can be represented by a special case of this model where $P_{G_i} = P_{B_i}$.

For the two state Markov chain, the steady state probability vector, of path p_i being in states *B* and *G*, π_i , is given by $\pi_i = \begin{bmatrix} \pi_{G_i} \\ \pi_{B_i} \end{bmatrix}$ where $\pi_{G_i} = \dfrac{\beta_i}{\alpha_i + \beta_i}$ and $\pi_{B_i} = 1 - \pi_{G_i}$.

The steady state loss probability on path *i*, denoted by $\overline{L}_i$, is given by:

$$\overline{L}_i = \lim_{t\to\infty} \Pr[\text{packet}\, t \text{ is lost on } p_i] = \pi_i^T B_i^1 , \qquad \textbf{(20)}$$

where π_i^T denotes the transpose of π_i .

[13] In many studies, such as [16], the values P_G=0 and P_B=1 are used, which leads to modeling bursts of consecutive losses. This definition fits for the E-model impairments modeling and the calculation of *R* value described in [6][7] [32].

4.2 The Noticeable Loss Rate Under Various dispersion strategies

4.2.1 The *NLR* under the *no-dispersion*

We start our analysis by first studying the *NLR* as observed over a single path. Let $L_i(t)$ be a random variable denoting the event of loss or success at time *t* on path p_i. Let $l_i(t)$ be the actual event occurring at *t* on p_i, $l_i(t) \in \{0,1,\phi\}$ where '1' denotes loss, '0' denotes success and ϕ denotes either loss or success ("don't care")[14].

Let $E_i(t,n) = (L_i(t),\cdots,L_i(t+n-1))$. For a particular event sequence $(l_i(t),\cdots,l_i(t+n-1))$ we want to compute $\Pr[E_i(t,n) = (l_i(t),\cdots,l_i(t+n-1))]$, which is done in the next theorem.

Theorem 1: Let $(l_i(t),\cdots,l_i(t+n-1))$ be an arbitrary success/loss sequence where $l_i(j) \in \{0,1,\phi\}$ $t \le j \le t+n-1$. Assume that the state probabilities at *t-1* are given by $\pi_i\ (t-1)$. Then:

$$\Pr[E_i(t,n) = (l_i(t),\cdots,l_i(t+n-1))] = \pi_i^T\,(t-1)\left(\prod_{j=0}^{n-1} \widehat{A}_i^{l_i(t+j)}\right)\mathbf{1} \qquad \textbf{(21)}$$

where $\widehat{A}_i^{l_i(t+j)} = \begin{cases} A_i\widehat{B}_i^1 & \text{if } l_i(t+j) = '1' \\ A_i\widehat{B}_i^0 & \text{if } l_i(t+j) = '0' \\ A_i & \text{if } l_i(t+j) = '\phi' \end{cases}$, $A_i = \begin{bmatrix} 1-\alpha_i & \alpha_i \\ \beta_i & 1-\beta_i \end{bmatrix}$, $\widehat{B}_i^1 = \begin{bmatrix} P_{G\,i} & 0 \\ 0 & P_{B_i} \end{bmatrix}$

$\widehat{B}_i^0 = \begin{bmatrix} 1-P_{G\,i} & 0 \\ 0 & 1-P_{B_i} \end{bmatrix}$, and where $\pi_i^T\,(t-1)$ denotes the transpose of the state probability vector at time *t-1*.

The proof is given in **Appendix A**.

Note that $\widehat{A}_i^{l_i(k)}$ denotes the matrix of probabilities where:

$\widehat{A}_i^{l_i(k)}(m,n) = \Pr[L_i(k) = l_i(k) \wedge S_i(k) = n \mid S_i(k-1) = m]$. That is, the $(m,n)^{th}$ entry is the probability for the Markov chain to transit from $S_i(t-1)$ to $S_i(t)$ and for packet *t* to be a loss/success/don't care, based on the value of $l_i(t)$.

[14] The actual event of cause is either '0' or '1'. The 'ϕ' event is modeled for cases where we do not care for the actual outcome of $l_i(t)$.

Theorem 1 can be used to compute the probability of any finite length arbitrary event sequence.

Remark 1: One should note the low complexity for computing (Eq. **(21**)). Despite the fact that the number of possible sequences is exponential in *n*, the special form of Eq. **(21**) allows to compute the probability of $E_i(t,n)$ in linear time in *n*.

4.2.1.1 **Calculating the *NLR***

Based on **(7**) and assuming that the state probability at *t-1* is given by $\pi_i^T(t-1)$, we may now compute the noticeable loss rate (based on the definition in **(4)**):

$$\Pr[NL_i^{(\delta)}(t)=1] = \Pr_i[l_i(t)=1] - \Pr_i[l_i(t)=1, l_i(t+1)=0, \cdots, l_i(t+\delta)=0] =$$

$$= \pi_i^T(t-1)B_{\mathrm{i}}^1 \text{ - } \pi_i^T(t-1)\left(\widehat{\mathrm{A}}_i^1\left(\widehat{\mathrm{A}}_i^0\right)^{\delta}\right)\mathbf{1} \ . \qquad \textbf{(22)}$$

When the system is under steady state we have $\pi_i(t) = \pi_i$, the noticeable loss rate, $NLR^{(\delta)}$, is given by:

$$NLR_i^{(\delta)} = \Pr[NL_{\mathrm{i}}^{(\delta)} = 1] = \pi_i{}^T B_{\mathrm{i}}^1 \text{ - } \pi_i{}^T\left(\widehat{\mathrm{A}}_i^1\left(\widehat{\mathrm{A}}_i^0\right)^{\delta}\right)\mathbf{1} \ . \qquad \textbf{(23)}$$

Now, assuming that each session takes a different path, that is, that without loss of generality session s_i takes path p_i, the expected network *NLR* for the *N* sessions, $\overline{NLR}^{(\delta)}$, is then:

$$\overline{NLR}_i^{(\delta)} = \frac{1}{N}\sum_{i=1}^{N}\left(\pi_i{}^T B_i^1 \text{ - } \pi_i{}^T\left(\widehat{\mathrm{A}}_i^1\left(\widehat{\mathrm{A}}_i^0\right)^{\delta}\right)\mathbf{1}\right). \qquad \textbf{(24)}$$

4.2.2 The *NLR* Under *Periodic Packet Dispersion*

Consider a *periodic dispersion* policy *Q*, with period length *K*. The policy is defined by $Q(k)$ ($Q(k) \in \{1 \ldots P\}$ and ($k = 1 \ldots K$)), meaning that packet *k* in the period will always be sent on $p_{Q(k)}$ periodically. Thus, the path taken for packet *t*, without loss of generality, is $p_{Q((t \bmod K)+1)}$.

To calculate the *NLR* we examine every path individually. In a matrix notation, when packet *t* is sent over $p_{Q(k)}$, its loss/success probabilities and state transition are obtained by multiplying the state vector of path $p_{Q(k)}$ at *t*-1, by $\widehat{A}_{Q(k)}^{l_{Q(k)}(t)}$ (where $\widehat{A}_{Q(k)}^{l_{Q(k)}(t)}$ is as defined in *Theorem* 1 above).

Now, consider packet *t* routed over $p_{Q(k)}$ and consider path $p_j \neq p_{Q(k)}$. For this path one must account for the state transition at time *t* but not for the probability of loss/success. We thus introduce the transition matrix T_i^Q for *periodic dispersion* policy *Q*, over path p_i:

$$T_i^Q[t] = \begin{cases} \widehat{A}_i^0 & \text{if } p_{Q((t \bmod K)+1)} = p_i \\ A_i & \text{otherwise} \end{cases}. \qquad \textbf{(25)}$$

First we want to compute the *NLR* for a session starting at time *t*. Assume that: $Q(t \bmod K) = k$, that is, the path selected at *t* is $p_{Q(k)}$. The *NLR* for this packet is obtained by calculating the probability that the packet is lost and subtracting the probability that the packet is lost and all subsequent δ packets arrive.

The latter probability is obtained by deriving the probability of the following event combination:

1) On path $p_{Q(k)}$ there is:
    i) A loss of packet *t*.
    ii) No loss on all packets $t+i$ $(i = 1...\delta)$, obeying $Q(((i+t) \bmod K)+1) = k$.
    iii) Don't care on all other packets.
2) On path $p_r \neq p_{Q(k)}$ there is:
    i) Don't care at packet *t*.
    ii) No loss on all packets $t+i$ $(i = 1...\delta)$, obeying $Q(((i+t) \bmod K)+1) = r$.
    iii) Don't care on all other packets.

This yields:

$$\Pr[\text{packet } t \text{ is NL}] = \pi_{Q(k)}^T B_{Q(k)}^1 - \left(\left(\pi_{Q(k)}^T \widehat{A}_{Q(k)}^1 \left(\prod_{i=1}^{\delta} T_{Q(k)}^Q[t+i] \right) \mathbf{1} \right) \prod_{\substack{r=1 \\ r \neq Q(k)}}^{P} \left(\pi_r^T \prod_{i=1}^{\delta} T_r^Q[t+i] \right) \mathbf{1} \right). \qquad \textbf{(26)}$$

By accounting for all possible starting positions in the period, the *NLR* for a session sent in a *periodic packet dispersion* policy *Q*, is then:

$$NLR^{(\delta)} = \frac{1}{K}\sum_{t=1}^{K}\left(\pi_{Q(k)}^{T} B_{Q(k)}^{1} - \left(\left(\pi_{Q(k)}^{T}\widehat{A}_{Q(k)}^{1}\left(\prod_{i=1}^{\delta} T_{Q(k)}^{Q}[t+i]\right)\mathbf{1}\right)\prod_{\substack{r=1\\ r\neq Q(k)}}^{P}\left(\pi_{r}^{T}\prod_{i=1}^{\delta} T_{r}^{Q}[t+i]\right)\mathbf{1}\right)\right) \quad \textbf{(27)}$$

Remark 2: Note that a straightforward analysis of the *P* path system may require taking a *P* dimensional state space, with computational complexity exponential in *P*. However, our analysis shows that the problem is decomposable and thus the computational complexity is only linear in *P*. The overall computational complexity is only: $O(P \cdot K \cdot \delta)$.

To calculate the average *NLR* for *N* sessions using the *periodic dispersion* strategy, note that the *NLR* experienced by s_i is not affected by session s_j. Therefore the expected $\overline{NLR}^{(\delta)}$ for *N* sessions over *P* paths is simply calculated by averaging *NLR* observed by the sessions.

4.2.3 The *NLR* Under *Deterministic Round Robin Dispersion*

A special case of *periodic dispersion* is the *deterministic round robin dispersion*, where the packets of a session are sent over paths $p_1 p_2 \cdots p_P$ in a round robin manner, starting at p_k. Clearly, Eq. **(27)** remains correct.

Consider a simple *round robin dispersion* policy *Q,* conducted over two paths p_1 and p_2, in which all odd packets are sent over p_1 and all even packets are sent over p_2.

Writing the probabilities implicitly, given the initial state probability vectors on the paths, $\pi_1(t-1)$ and $\pi_2(t-1)$, we have:

$$\begin{aligned}
\Pr[NL^{(\delta)}(t)=1] = {} & \Pr[\text{session starts at } p_1]\cdot(\Pr[l_1(t)=1] - \\
& \Pr[E_1(t,\delta+1)=(1,\phi,0,\ldots,\phi,0) \wedge E_2(t,\delta+1)=(\phi,0,\phi,\ldots,0,\phi)]) \\
& + \Pr[\text{session starts at } p_2]\cdot(\Pr[l_2(t)=1] - \\
& \Pr[E_2(t,\delta+1)=(1,\phi,0,\ldots,\phi,0) \wedge E_1(t,\delta+1)=(\phi,0,\phi,\ldots,0,\phi)]) \quad \textbf{(28)}
\end{aligned}$$

where ϕ stands for a “don’t care”. Cleary Eq. **(28)** is a special case of Eq. **(27)**. The *NLR* for the system, assuming steady state and even δ, is then:

$$NLR^{(\delta)} = \frac{1}{2}\left(\pi_1^T B_1^1 - \left(\pi_1^T \left(\widehat{A}_1^1 \left(A_1 \widehat{A}_1^0\right)^{\delta/2}\right)\mathbf{1}\right)\left(\pi_2^T \left(\left(A_2 \widehat{A}_2^0\right)^{\delta/2}\right)\mathbf{1}\right)\right)$$
$$+\frac{1}{2}\left(\pi_2^T B_2^1 - \left(\pi_2^T \left(\widehat{A}_2^1 \left(A_2 \widehat{A}_2^0\right)^{\delta/2}\right)\mathbf{1}\right)\left(\pi_1^T \left(\left(A_1 \widehat{A}_1^0\right)^{\delta/2}\right)\mathbf{1}\right)\right) \quad \text{.} \qquad \textbf{(29)}$$

The derivation for odd δ is similar.

4.2.4 The *NLR* Under *Random Packet Dispersion*

In our analysis we assume that the loss models over the paths are independent, meaning that the state ($S_i(t)$) on path p_i is independent of the state ($S_j(t)$) on path $p_j \ \forall j \neq i$, at time *t.* A session dispersed over the paths using the *random dispersion strategy*, experiences losses as if it was delivered over a single path with the underlying loss model that is the combination of loss models over the paths. We will denote the "equivalent path" by p'.

First we calculate the characteristics of p' as observed by the session. The "equivalent path" is characterized by 2^P states, resulting from the cross product of the states of the individual paths, $S' = \{S_1 \times S_2 \times \cdots \times S_R\}$. For the sake of clarity one may enumerate and index the states in S' as $1,2,...2^P$. Let $\Gamma(2^P \times 2^P)$ denote the transition matrix for p' where the $(m,n)^{th}$ entry is the probability that p' moves at time *t+1* to state *n,* given that is was at state *m* at time *t.*

Matrix Γ is a transition matrix for a Markov chain with 2^P states. This Markov chain has a steady state probability vector denoted by $\hat{\pi}$, and can be easily computed from the steady states over the individual paths by noting that $P[(S_1(t),...,S_P(t)) = (x_1,...,x_P)] = \prod_{i=1}^{P} P[S_i(t) = x_i]$, where $P[S_i(t) = x_i]$ is the steady state probability of being in state x_i, on path p_i.

In *random dispersion* the decision regarding over which path to send packet *t* of session s_q, is done in a random fashion. Let $\rho_{q,i}$ ($\sum_i^P \rho_{q,i} = 1$) denote the probability that packets of s_q are sent on p_i.

Similar to the expression received for the *no-dispersion* (see **(22)**), given the initial state vector $\hat{\pi}(t-1)$, we get the following expression for *NLR* under *random dispersion* for session s_q:

$$NLR^{(\delta)} = \hat{\pi}^T(t-1)\hat{B}\mathbf{1} - \hat{\pi}^T(t-1)\left(\Gamma\hat{B}^1\left(\Gamma\hat{B}^0\right)^{\delta}\right)\mathbf{1} \quad \textbf{(30)}$$

where, (as in *Theorem* 1) Γ denotes the matrix of transition probabilities: $\Gamma(m,n) = \Pr[S_i'(t) = n \mid S_i'(t-1) = m]$, and where $\hat{B}^1$ is a $2^P x 2^P$ diagonal matrix, whose (*m*,*m*)th entry ($1 \le m \le 2^P$), is the loss probability in state *m*, calculated as in the example in the next section (4.2.4.1), as follows: Assuming that state *m* is given by $(S_1,...,S_P)$ where $S_i \in \{\mathrm{G},\mathrm{B}\}$, then this entry is given by $\sum_{i=1}^{P} \rho_{q,i}(P_{B_i} I(S_i = B) + P_{G_i} I(S_i = G))$, where $I(\ldots)$ is the indicator function. Similarly, the (*m*,*m*)th entry of $\hat{B}^0$ is given by $\sum_{i=1}^{P} \rho_{q,i}((1-P_{B_i})I(S_i = B) + (1-P_{G_i})I(S_i = G))$, that is $\hat{B}^0 = \mathbf{I} - \hat{B}^1$, where $\mathbf{I}(2^P x 2^P)$ is a unit matrix.

Remark 3: Note that the computation complexity is exponential in the number of paths: $O(2^P \cdot \delta)$.

4.2.4.1 A Special case: NLR Under Random Packet Dispersion for two paths

In this section we present an example for the *NLR* computation for the *random dispersion* strategy. The results from this section will be used in Section 4.3.

Consider a simple *random dispersion* strategy where packets of s_q, are sent in probability of $\rho_{q,1}$ and $\rho_{q,2}$, over two paths p_1 and p_2 accordingly. The state space over the combined path observed by the sessions is $S' = \{GG, GB, BG, BB\}$, where, for example, *GB* stands for *G* in path 1 and *B* in path 2, etc.

The transition matrix in the this Markov chain is:

$$\Gamma = \begin{pmatrix} (1-\alpha_1)(1-\alpha_2) & (1-\alpha_1)\alpha_2 & \alpha_1(1-\alpha_2) & \alpha_1\alpha_2 \\ (1-\alpha_1)\beta_2 & (1-\alpha_1)(1-\beta_2) & \alpha_1\beta_2 & \alpha_1(1-\beta_2) \\ \beta_1(1-\alpha_2) & \beta_1\alpha_2 & (1-\beta_1)(1-\alpha_2) & (1-\beta_1)\alpha_2 \\ \beta_1\beta_2 & \beta_1(1-\beta_2) & (1-\beta_1)\beta_2 & (1-\beta_1)(1-\beta_2) \end{pmatrix}.$$

The steady state $\hat{\pi}$ is: $\hat{\pi} = \begin{bmatrix} \pi_{GG} \\ \pi_{GB} \\ \pi_{BG} \\ \pi_{BB} \end{bmatrix} = \begin{bmatrix} \pi_{G_1}\pi_{G_2} \\ \pi_{G_1}\pi_{B_2} \\ \pi_{B_1}\pi_{G_2} \\ \pi_{B_1}\pi_{B_2} \end{bmatrix}$

$$\hat{B}^1 = \begin{bmatrix} \rho_{q,1}P_{G_1} + \rho_{q,2}P_{G_2} & 0 & 0 & 0 \\ 0 & \rho_{q,1}P_{G_1} + \rho_{q,2}P_{B_2} & 0 & 0 \\ 0 & 0 & \rho_{q,1}P_{B_1} + \rho_{q,2}P_{G_2} & 0 \\ 0 & 0 & 0 & \rho_{q,1}P_{B_1} + \rho_{q,2}P_{B_2} \end{bmatrix}$$

and $\hat{B}^0 = \mathbf{I} - \hat{B}^0$, where $\mathbf{I}(2^2 x 2^2)$ is a unit matrix.

The *NLR* experienced by the session in steady state is then:

$$NLR^{(\delta)} = \hat{\pi}^T \hat{B}^1 \mathbf{1} - \hat{\pi}^T \left(\Gamma \hat{B}^1 \left(\Gamma \hat{B}^0 \right)^{\delta} \right) \mathbf{1} \ . \qquad \textbf{(31)}$$

4.2.4.2 NLR Under Random Packet Dispersion for N sessions over P paths

To calculate the average *NLR* for *N* sessions using the *random dispersion* strategy over *P* paths we define $\hat{B}_q^1$ for session s_q, $q=1,\dots,N$, as in Eq. **(30)** (that is, the *(m,m)*th entry is given by $\sum_{i=1}^{P} \rho_{q,i} (P_{B_i} I(s_i = B) + P_{G_i} I(s_i = G))$ and similarly $\hat{B}_q^0$. The average *NLR* experienced by the sessions, denoted by $\overline{NLR}^{(\delta)}$, is then:

$$\overline{NLR}^{(\delta)} = \frac{1}{N} \sum_{q=1}^{N} \left(\hat{\pi}^T (t-1) \hat{B}_q^1 \mathbf{1} - \hat{\pi}^T (t-1) \left(\Gamma \ \hat{B}_q^1 \left(\Gamma \ \hat{B}_q^0 \right)^{\delta} \right) \mathbf{1} \right) \qquad \textbf{(32)}$$

Note that proper planning the $\rho_{q,i}$ can highly reduce the average *NLR* and thus enhance quality.

4.3 Comparison of the Dispersion Strategies Under Gilbert loss model

In this section we compare the *NLR* experienced by sessions sent via dispersion strategies over paths experiencing underlying Gilbert loss model. Since the loss model is affected by four parameters, it is difficult to present a thorough comparison. For simplicity we will compare paths with equal characteristics and will assume that in all paths $P_{G_i} = 0$. A numerical comparison of paths with different characteristics leads to similar conclusions.

4.3.1 Translating results to MOS

To emphasis the significant gain that can be achieved by implementing packet dispersion strategies, consider the following results, given in Table 1, for the case where for $\delta = 2$ $P_B = 0$, based on the results in Figure 1:

	Model parameters	***No-dispersion***		***Round Robin Dispersion***		***Random Dispersion***		***Average Loss Rate***	***Maximal gain***	
		NLR	***MOS***	***NLR***	***MOS***	***NLR***	***MOS***		***NLR***	***MOS***
1	$T_B = 100$, $T_G = 1000$, $P_B = 0.5$	3.36%	~4.1	2.35%	~4.3	2.09%	~4.4	4.55%	1.27%	**~0.3**
2	$T_B = 200$, $T_G = 1000$, $P_B = 0.4$	4.24%	~4	2.92%	~4.2	2.69%	~4.3	7.27%	1.55%	**~0.3**
3	$T_B = 200$, $T_G = 1000$, $P_B = 0.5$	6.27%	~3	4.49%	~3.8	4.06%	~4	9.09%	2.21%	**~0.8**

Table 1: The affect of dispersion strategies on MOS.

In Table 1 we can see that packet dispersion over two identical paths can improve the MOS in up to approximately one grade, which is a huge improvement that elevates perceived quality from "poor" to "acceptable" or "good". In Table 1 we show extreme results in which the improvement is meaningful, but in most cases we will see that though packet dispersion can reduce *NLR* the total affect on quality might be small. Clearly, there are other examples in which packet dispersion reduces quality under other network characteristics as will be presented in the following examples.

For a better understanding of the results we present plots comparing differences and ratios between the strategies given in figures 10-31. In the plots we present the Markov chain parameters in term of T_G and T_B, which are the average duration time for the chain to be in states *G* and *B* accordingly ($T_G = 1/\alpha$, $T_B = 1/\beta$). The duration time in our model is actually measured in the number of packets sent in each state (i.e. $T_G = 100$ means that 100 packets are sent in average in *G* state, for packetization periods of 30ms in codecs this would mean 3 seconds).

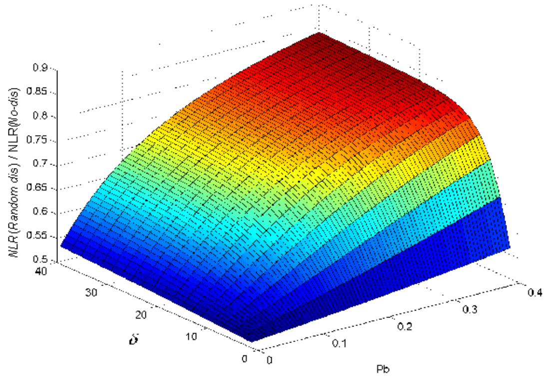
In general we may conclude that under a vast range of network conditions, packet dispersion, both *random dispersion* and *periodic dispersion*, can highly reduce the *NLR* in comparison to the traditional *no-dispersion* strategy. We also showed that there are occasions that bring opposite results.

Remark 4: In the comparisons we can see that the largest differences between the strategies are when $2 \le \delta \le 10$. The reason for that is that we compare the strategies using two paths only. Clearly, if more paths are used in the dispersion the range of δ will grow, and thus have greater effect on quality. Note that ff *FEC* or *PLC* mechanisms are used, $\delta = 2$ is a reasonable value and therefore the results are meaningful in many practical cases.

We could not find any simple rules of thumb for the decision as to which dispersion strategy to choose under specific network characteristics. This is because the parameters affect each other when calculating the *NLR*. As a general direction we may say that if $T_B << T_G$, $P_G \cong 0$, $P_B << 1$ and $\delta \cong P$ (where *P* is the number of paths available for dispersion) packet dispersion can improve VoIP perceived quality.

4.3.2 *Random Dispersion* vs. *no-dispersion*

Here we compare a session that is subject to *no-dispersion* to a session that is subject to *random dispersion* over two equal paths where half of the packets are sent over p_1 and half over p_2. In figures 11-15 we plot the differences and the ratios between the two dispersion strategies as a function of P_B, δ and fixed T_G and T_B. We can see that the *random dispersion* can reduce *NLR* in up to 50% and the nominal reduction can reach up to more than 1%. But under other conditions (figures 14-15) *no-dispersion* may result with lower *NLR*. Thus we conclude that the advantage of *random dispersion* over *no-dispersion* is not absolute when the paths experience Gilbert losses.



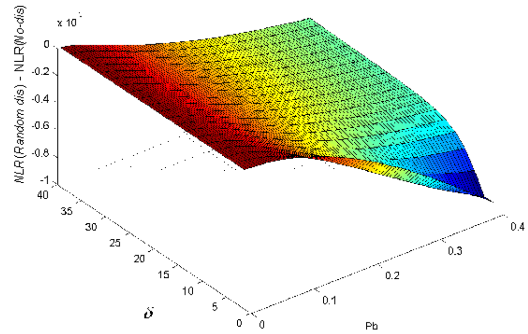
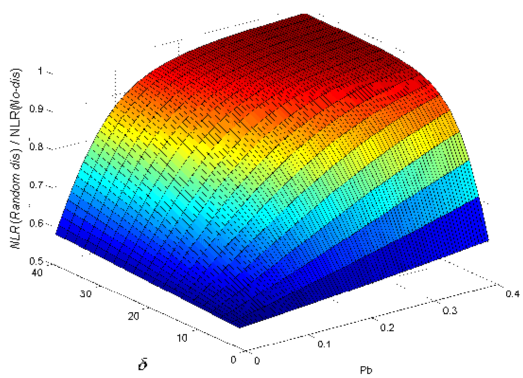


Figure 10: ***NLR*** **ratio between** ***random dispersion*** **and** ***no-dispersion*** **for** $T_G = 1000$ **and** $T_B = 10$

Figure 11: ***NLR*** **difference between** ***random dispersion*** **and** ***no-dispersion*** **for** $T_G = 1000$ **and** $T_B = 10$



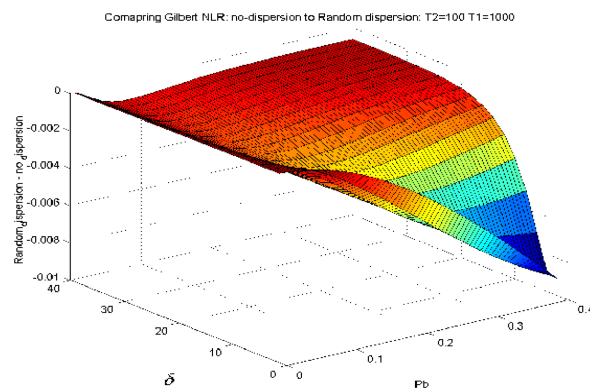
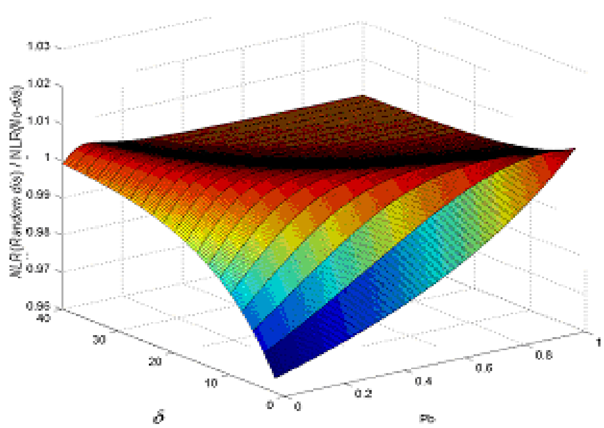


Figure 12: ***NLR*** **ratio between** ***random dispersion*** **and** ***no-dispersion*** **for** $T_G = 1000$ **and** $T_B = 100$

Figure 13: ***NLR*** **difference between** ***random dispersion*** **and** ***no-dispersion*** **for** $T_G = 1000$ **and** $T_B = 100$



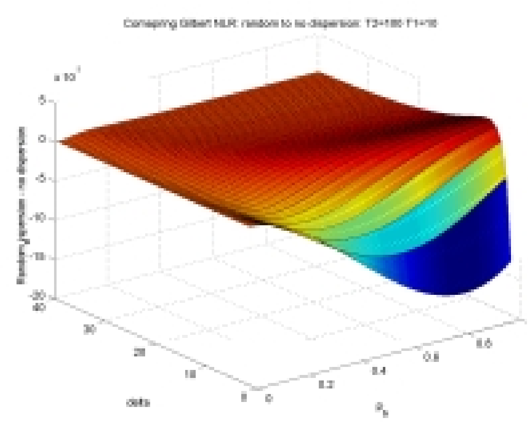


Figure 14: ***NLR*** **ratio between** ***random dispersion*** **and** ***no-dispersion*** **for** $T_G = 10$ **and** $T_B = 100$

Figure 15: ***NLR*** **difference between** ***random dispersion*** **and** ***no-dispersion*** **for** $T_G = 10$ **and** $T_B = 100$

In Figures 16-17 we plot the *NLR* difference and ratio between *random dispersion* and *no-dispersion* for fixed $T_G = 1000$ and $\delta = 2$. We can see that the gain of *random dispersion* can reach up to 50% but the nominal reduction of the *NLR* at that point is low. A meaningful reduction (more than 6%) of *NLR* occurs when the losses in the *Bad* state are frequent. This result is quite intuitive; the loss bursts are expected to be spread and become unnoticeable since loss distance is small ($\delta = 2$). Take for example $T_B = 200$ and $P_B = 0.6$, the average loss rate is 10%, the *NLR* reduction is approximately 4%, which highly improves quality.

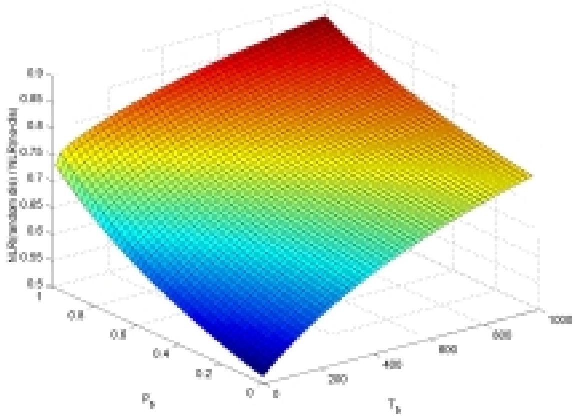


Figure 16: ***NLR*** **ratio between** ***random dispersion*** **and** ***no-dispersion*** **for** $T_G = 1000$ **and** $\delta = 2$

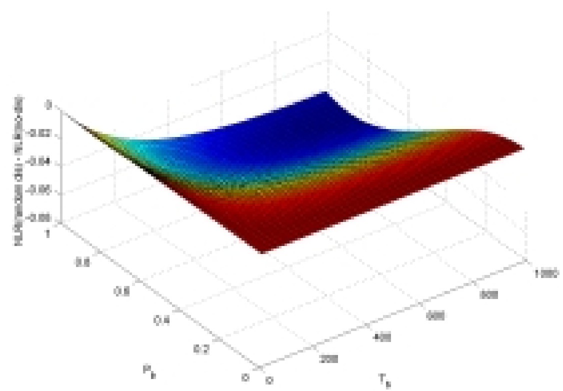


Figure 17: ***NLR*** **difference between** ***random dispersion*** **and** ***no-dispersion*** **for** $T_G = 1000$ **and** $\delta = 2$

4.3.3 *Round Robin Dispersion* vs. *no-dispersion*

Here we compare a session that is subject to *no-dispersion* to a session that is subject to *round robin deterministic dispersion* over two equal paths where half of the packets are sent over p_1 and half over p_2. In figures 18-23 we plot the differences and the ratios between the two dispersion strategies as a function of P_B, δ and fixed T_G and T_B. We can see that the *round robin dispersion* can reduce *NLR* in up to 50% and the nominal reduction can reach up to more than 1%. But under other conditions (figures 22-23) *no-dispersion* may result with lower *NLR*. Thus we conclude that the advantage of *round robin dispersion* over *no-dispersion* is not absolute when the paths experience Gilbert losses.

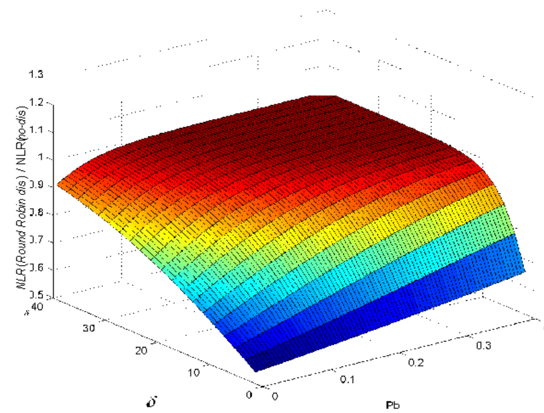


Figure 18: ***NLR*** **ratio between** ***round robin dispersion*** **and** ***no-dispersion*** **for** T_G=1000 **and** T_B=10

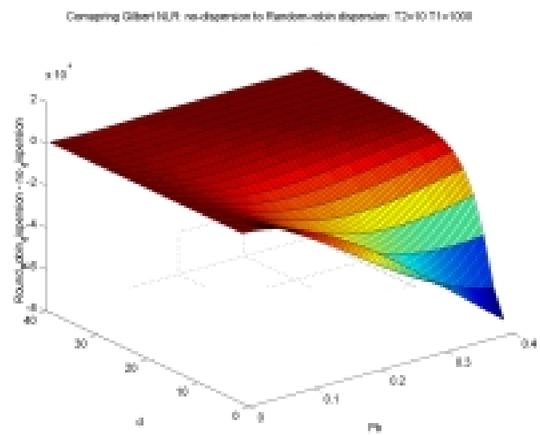


Figure 19: ***NLR*** **difference between** ***round robin dispersion*** **and** ***no-dispersion*** **for** T_G=1000 **and** T_B=10

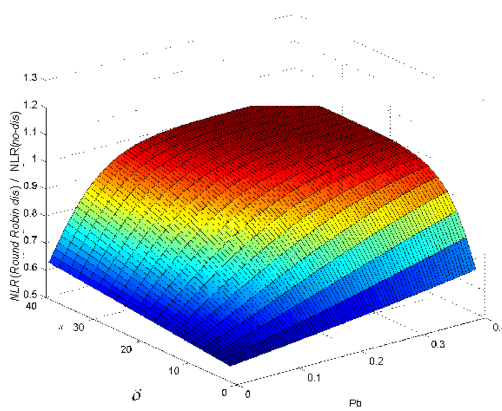


Figure 20: ***NLR*** **ratio between** ***round robin dispersion*** **and** ***no-dispersion*** **for** T_G=1000**,** T_B=100

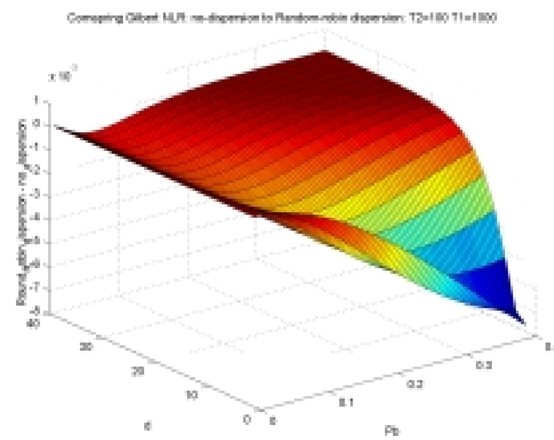


Figure 21: ***NLR*** **difference between** ***round robin dispersion*** **and** ***no-dispersion*** **for** T_G=1000 **and** T_B=100

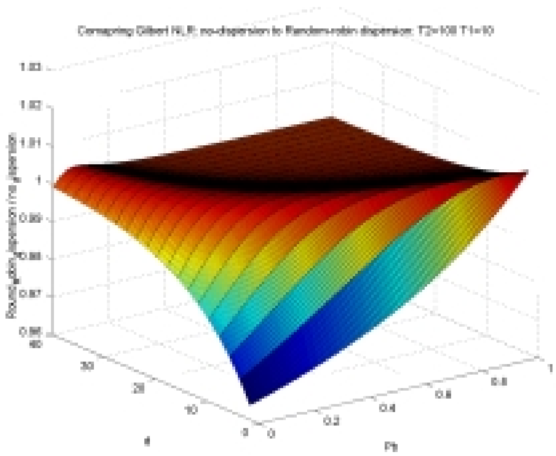


Figure 22: ***NLR*** **ratio between** ***round robin dispersion*** **and** ***no-dispersion*** **for** T_G=10 **and** T_B=100

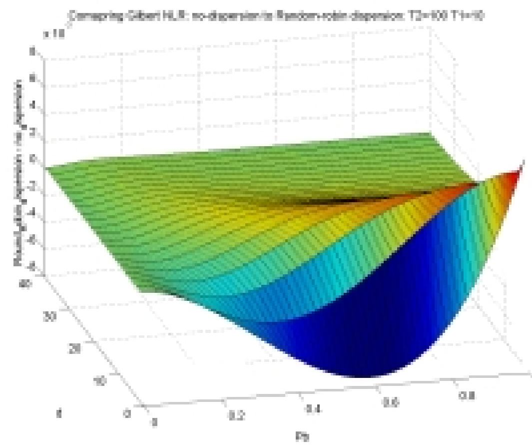


Figure 23: ***NLR*** **difference between** ***round robin dispersion*** **and** ***no-dispersion*** **for** T_G=10 **and** T_B=100

In Figures 24-25 we plot the *NLR* difference and ratio between *round robin dispersion* and *no-dispersion* for fixed $T_G = 1000$ and $\delta = 2$. We can see that the gain of *round robin dispersion* can reach up to 50%, since the losses are spread between the sessions, but the nominal reduction of the *NLR* at that point is very low. A meaningful reduction of *NLR* (up to 4%) occurs when the loss probability in the *Bad* state is 50%. Note that unlike in the *random dispersion* case, when the loss rate in the *Bad* state become higher than 50%, the efficiency of the round *robin dispersion* decreases. This happen since a session that experiences a loss at the *Bad* state on one link will continue experiencing the losses on that link that are within the loss distance and therefore will be counted as noticeable losses.

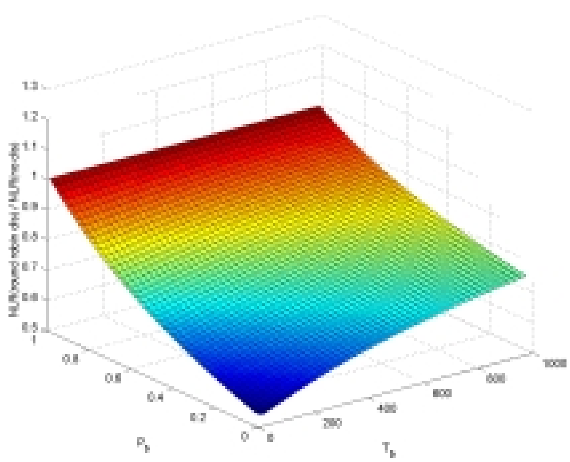


Figure 24: ***NLR*** **ratio between** ***round robin dispersion*** **and** ***no-dispersion*** **for** T_G=1000, $\delta = 2$

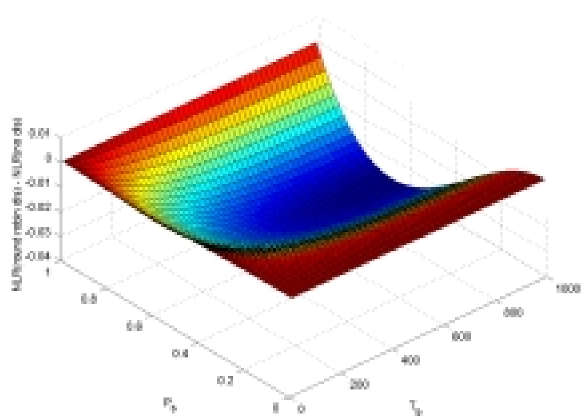


Figure 25: ***NLR*** **difference between** ***round robin dispersion*** **and** ***no-dispersion*** **for** T_G=1000**,** $\delta = 2$

4.3.4 *Random Dispersion* vs. *Round Robin Dispersion*

Here we compare a session that is subject to *random dispersion* to a session that is subject to *round robin deterministic dispersion* over two equal paths where half of the packets are sent over p_1 and half over p_2. In figures 26-31 we plot the differences and the ratios between the two dispersion strategies as a function of P_B, δ and fixed T_G and T_B.

In most of Gilbert model scenarios we analyzed *random dispersion* resulted in lower *NLR* but the differences are small. We can see that the in some cases *random dispersion* can reduce *NLR* in up to 40% in comparison to *deterministic dispersion* but the nominal reduction is very low in these particular examples (figures 26-29). Under other conditions (figures 30-31) *no-dispersion* may result with lower *NLR*.

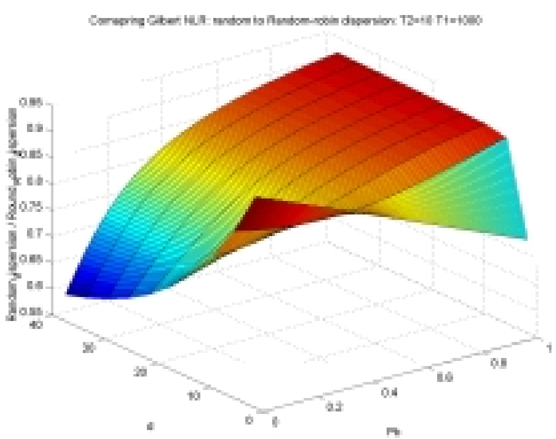


Figure 26: *NLR* ratio between *random* and *round robin dispersion* for $T_G = 1000$ and $T_B = 10$

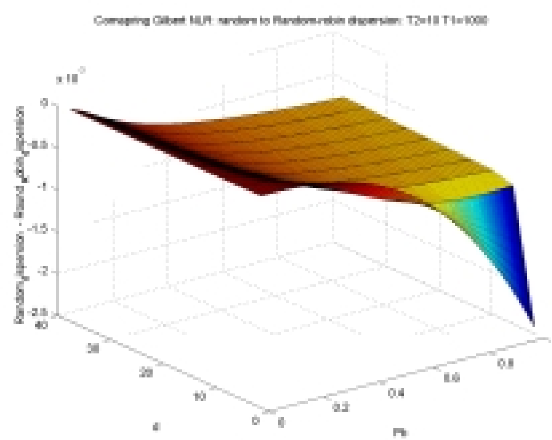


Figure 27: *NLR* difference between *random* and *round robin dispersion* for $T_G = 1000$ and $T_B = 10$

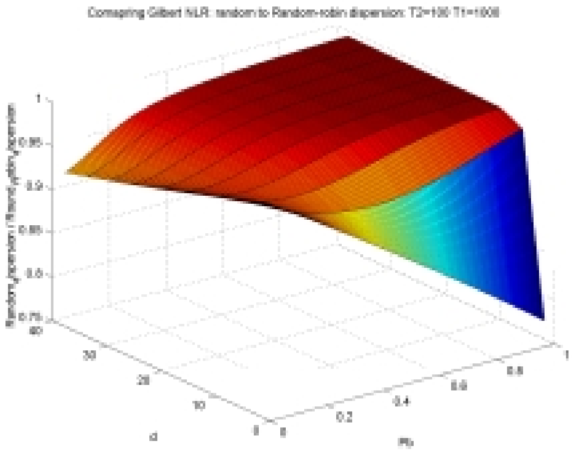


Figure 28: ***NLR*** **ratio between** ***random dispersion*** **and** ***round robin*** **for** $T_G = 1000$ **and** $T_B = 100$

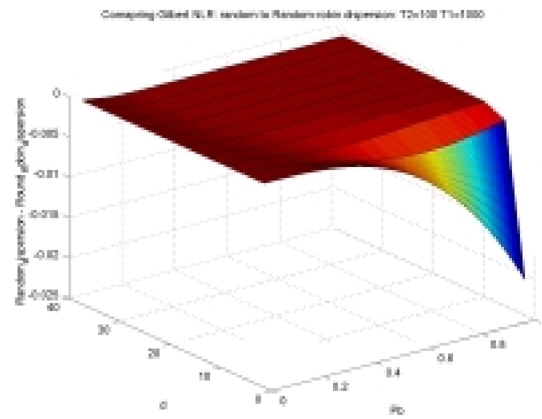


Figure 29: ***NLR*** **difference between** ***random dispersion*** **and** ***round robin*** **for** $T_G = 1000$ **and** $T_B = 100$

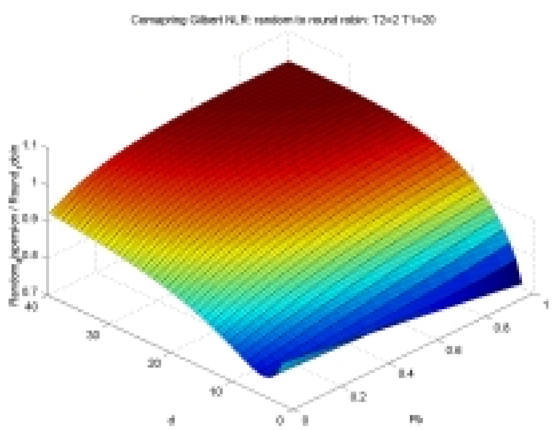


Figure 30: ***NLR*** **ratio between** ***random dispersion*** **and** ***round robin*** **for** $T_G = 20$ **and** $T_B = 2$

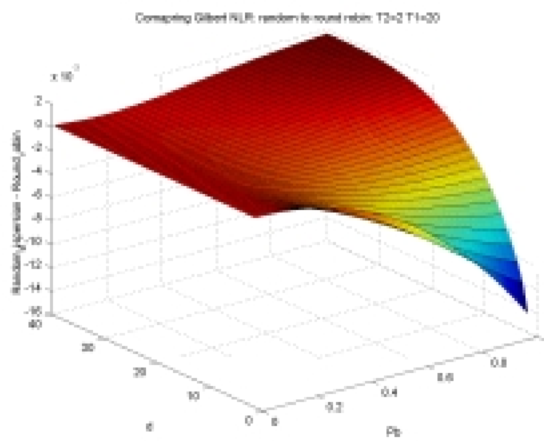


Figure 31: ***NLR*** **difference between** ***random dispersion*** **and** ***round robin*** **for** $T_G = 20$ **and** $T_B = 2$

In Figures 32-33 we plot the *NLR* difference and ratio between *round robin dispersion* and random *dispersion* for fixed $T_G = 1000$ and $\delta = 2$. We can see that the gain of *round robin dispersion* can reach up to 25% and up to nominal reduction of more than 6%, since the losses are better spread between the sessions, as can be deduced from Figures 25 and 17.

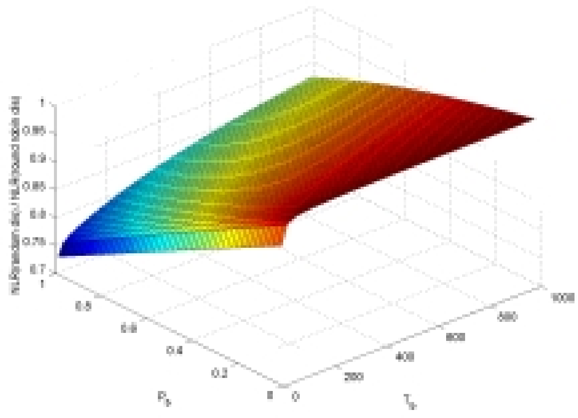


Figure 32: ***NLR*** **ratio between** ***random dispersion*** **and** ***round robin dispersion*** **for** $T_G = 1000$ **,** $\delta = 2$

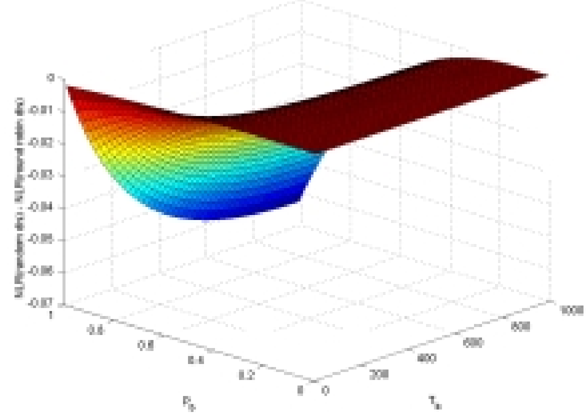


Figure 33: ***NLR*** **difference between** ***random dispersion*** **and** ***round robin dispersion*** **for** $T_G = 1000$ **,** $\delta = 2$

5 Conclusions

We addressed the factors affecting voice quality of VoIP and focused on packet loss. We proposed the *noticeable loss rate (NLR)* as a metric well correlated with voice quality for VoIP applications. We studied the effect of packet dispersion strategies, as performed de-facto by load balancing (multi-homing) devices or can be implemented using other mechanisms, on the *NLR*. We conducted this analysis under the assumption of Bernoulli losses and the Gilbert loss model, over the network paths.

We showed that under the Bernoulli loss model, in most practical cases the discussed packet dispersion strategies could reduce *NLR* and thus improve voice quality. We showed that for similar paths all dispersion strategies and *no-dispersion* are equally good and thus packet dispersion provides no major benefit in terms of reducing the *NLR*. We also showed that *random dispersion* is superior to *periodic dispersion* (under several assumptions) and as such are preferred for VoIP applications.

We provided mathematical analysis of the *NLR* for sessions traveling over paths experiencing bursty loss model (Gilbert model). We provided low complexity expressions for the computation of the *NLR* under the dispersion strategies. We demonstrated in numerical examples that the effectiveness of the various packet dispersion strategies strongly depends on the model parameters, but in many environments both *periodic dispersion* and *random dispersion* can highly reduce *NLR* in comparison to the traditional routing, where a single path is used.

6 Future study

We have studied the effect of various packet dispersion strategies on links experiencing Random losses or Gilbert-model losses. The effect of dispersion on other loss models, such as extended Gilbert model (Markov chain with three states: *Good*, *Bad*, *Failure*, and the general *N* state Markov chain) is yet to be studied.

This study focused on the effect of dispersion on losses. Accounting for both loss and delay in one model is another subject for future study.

We have mentioned in this study the multi-homing (load-balancing) devices as a possible practical cause for packet dispersion in IP networks. The proliferation of multi-homing devices raises some interesting questions regarding future network performance. Traditional networks are based on the fundamental property that traffic is routed along single routes. Packet dispersion, implemented de facto by network devices such as load-balancers, introduces a dramatic change in this principle and thus may fundamentally change the network operations and behavior (we will refer to a network where multiple nodes disperse packets as *dispersed network*). Due to this change the operation principles of modern computer networks need to be re-visited and re-examined in the light of packet dispersion. The major question is then what would be *dispersed network* behavior and how would applications be affected in light of this fundamental change in the network operations.

When discussing network behavior we must first address the issue of routing in an environment where nodes make independent dispersion decisions. Such environment will create disjoint or partially disjoint routes and may lead to routing loops. Thus, routing algorithms must take packet dispersion into account to avoid loops and choose routes that achieve best performance (e.g. choosing routes with similar delay can be important). The end-to-end network characteristics of *dispersed networks*, in terms of loss delay and jitter, and their affect on applications is an issue that may affect network end users sooner than we expect.

References

[1] D. G. Apostolopoulos. “Reliable video communication over lossy packet networks using multiple state encoding and path diversity. In *Proceeding Visual Communication and Image Processing, pages 392-409,* Jan. 2001.

[2] E. Altman, K. Avrachenkov, C. Barakat, “TCP in Presence of Bursty Losses”, Proceedings of ACM SIGMETRICS, Santa Clara, California, June 2000.

[3] Y. Birk and N. Bloch, “Improving network performance with prioritized dispersal”.

[4] J-C. Bolot, “End-to-end packet delay and loss behavior in the Internet”, In Proceedings of ACM SIGCOMM '93 Conference on Communications Architectures, Protocols and Applications, pages 289{298, San Francisco, CA, USA,September 1993.

[5] M-S. Borella, D. Swider, S. Uludag, G-B. Brewster, “Internet Packet Loss: Measurement and Implications for End-to-End QoS”, International Conference on Parallel Processing, August 1998.

[6] A. D. Clark, “Modeling the effect of Burst Loss and Recency on Subjective Voice Quality”, Proc. IP Telephony Workshop 2001.

[7] ETSI – DRAFT TS 101 329 – 5 V1.1.2 (2001-08)\ Q.

[8] Hung Ngo, S. Varadarajan and J. Srivastava, “Error-Spreading: Reducing Bursty Error in Continuous Media streaming”, in Proceedings of *IEEE Multimedia Systems '99* (ICMCS), vol. 1 1999: 314-319.

[9] F. Ishizaki, “Study on reduction of total bandwidth requirement by traffic dispersion”, Proceedings of International Conference on ATM and High Speed Internet 2001, pp.285-289, Seoul, Korea, 2001.

[10] ITU-T, “The E-model, a computational model for use in transmission planning”, recommendation G.107.

[11] ITU-T, “A high quality low complexity algorithm for packet loss concealment with G.711”, recommendation G.711 Appendix I (1999).

[12] ITU-T, “Transmission Impairments”, recommendation G.113, 02/96.

[13] ITU-T, “Objective quality assessment of telephone band speech codecs (PSQM)”, recommendation P.861.

[14] ITU-T, “Perceprual Evaluation of Speech Quality (PESQ)”, Recommendation P.862.

[15] W. Jiang and H. Schulzrinne. “Qos measurement of internet real-time multimedia services”, Technical Report CUCS-015-99, Columbia University, December 1999.

[16] W. Jiang and H. Schulzrinne, “Modeling of Packet Loss and Delay and Their Effect on Real-Time Multimedia Service Quality”, NOSSDAV 2000 Paper #27.

[17] W. Jiang and H. Schulzrinne. ”Comparison and Optimization of Packet Loss Repair Methods on VoIP Perceived Quality under Bursty Loss”, NOSSDAV 2002.

[18] W. Jiang, H. Schulzrinne, “Perceived Quality of Packet Audio under Bursty Losses”, IEEE INFOCOM 2002.

[19] R. Koodli, R. Ravikanth “One-Way Loss Pattern Sample Metrics”, ietf, draft-ietf-ippm-loss-pattern-05.txt, 1999.

[20] Yi J. Liang, and B. Girod, “A Simulation Study of Packet Path Diversity for TCP File Transfer and Media”, Proceedings of 2002 Tyrrhenian International Workshop on Digital Communications (IWDC 2002), Capri, Italy, Sept. 2002.

[21] Yi J. Liang, E-G. Steinbach and B. Girod, “Multi-Stream Voice Over IP Using Packet Path Diversity”, Proceedings IEEE Fourth Workshop on Multimedia Signal Processing, pp. 555-560, Cannes, France, Oct. 2001.

[22] Yi J. Liang, E-G. Steinbach and B. Girod, “Real-time Voice Communication over the Internet Using Packet Path Diversity”, ACM Multimedia '01, Ottawa, Canada, Oct. 2001.

[23] A. P.Markopoulou, F-A. Tobagi, M-J. Karam, “Assessment of VoIP quality over Internet Backbones”, In Proceedings of IEEE Infocom, June 2002.

[24] V. Markovski, “Simulation and Analysis of Loss in IP networks”, Thesis, University “Sts. Cyril and Methodius", Macedonia, October 2000.

[25] Y. Nebat and M. Sidi, “Resequencing Considerations in Parallel Downloads”, INFOCOM 2002.

[26] T. Nguyen and A. Zakhor, “Path Diversity with Forward Error Correction (PDF) System for Packet Switched Networks”, INFOCOM 2003, April 1-5, San Francisco CA, USA.

[27] F. Qi, J. Q. Mei, D. Xia, S. Xu, “New proofs of weighted power mean inequalities and monotonicity for generalized weighted mean values”, Mathematical Inequalities and

Applications 3 (2000), no.3, 377--383. RGMIA Research Report Collection 2 (1999), no.1, Article 10, 99--102; 2 (1999), no.4, Article 6, 469--472.

[28] J. Rosenberg, L. Qiu, H. Schulzrinne, “Integrating Packet FEC into Adaptive Voice Playout Buffer Algorithms on the Internet”, INFOCOM 2000.

[29] H. Schulzrinne, S-L. Casner, R. Frederick, V. Jacobson, “RTP: a transport protocol for real-time applications”, IETF, RFC 1889, 1996.

[30] L.F Sun, G. Wade, B. M. Lines, E. C. Ifeachor, “Impact of Packet Location on Perceived Speech Quality”, Proceedings of 2nd IP-Telephony Workshop (IPTEL '01), Columbia University, New York, pp114-122, 2-3 April, 2001.

[31] D. Thaler, C. Hopps, “Multipath Issues in Unicast and Multicast Next-Hop selection”, RFC2991, Novemver 2000.

[32] TIA, “Telecommunications IP Telephony Equipment Voice Quality Recommendations for IP Telephony” – PN-4689 (TIA/EIA/TSB116) TIA Engineering Committee TR-41.1.

[33] S-J. Yang, “Traffic Dispersion For Bursty Traffic On Heterogeneous Networks”, Master Thesis Report, 1998, UT Austin.

[34] L. Zhong, “A Binary Channel with Additive Burst Noise Based on a Finite Queue”, Thesis, Queen’s University, Kingstone, Ontario, Canada, November 2000.

Appendix A

Proof: We will prove the following two claims for $n \geq 1$.

1) $$\Pr[E_i(t,n) = (l_i(t),\cdots,l_i(t+n-1)) \wedge S_i(t+n-1) = G] = \pi_i^T(t-1)\left(\prod_{j=0}^{n-1} \widehat{A}_i^{l_i(t+j)}\right)[1] \quad \textbf{(33)}$$

2) $$\Pr[E_i(t,n) = (l_i(t),\cdots,l_i(t+n-1)) \wedge S_i(t+n-1) = B] = \pi_i^T(t-1)\left(\prod_{j=0}^{n-1} \widehat{A}_i^{l_i(t+j)}\right)[2] \quad \textbf{(34)}$$

The proof of these claims is carried out by induction on both claims concurrently.

<u>Induction basis (*n=1*):</u> the proof of the basis is immediate from the definitions:

Note that starting at *t-1*, a state transition is performed and based on the state at *t*, the loss/success event occurs.

$$\Pr[l_i(t) = '1' \wedge S_i(t) = B] = (\Pr[S_i(t-1) = G]\cdot\alpha + \Pr[S_i(t-1) = B]\cdot(1-\beta))\cdot P_{B_i}$$
$$\Pr[l_i(t) = '1' \wedge S_i(t) = G] = (\Pr[S_i(t-1) = G]\cdot(1-\alpha) + \Pr[S_i(t-1) = B]\cdot\beta)\cdot P_{G_i}$$

$$\Pr[l_i(t) = '0' \wedge S_i(t) = B] = (\Pr[S_i(t-1) = G]\cdot\alpha + \Pr[S_i(t-1) = B]\cdot(1-\beta))\cdot(1-P_{B_i})$$
$$\Pr[l_i(t) = '0' \wedge S_i(t) = G] = (\Pr[S_i(t-1) = G]\cdot(1-\alpha) + \Pr[S_i(t-1) = B]\cdot\beta)\cdot(1-P_{G_i})$$

$$\Pr[l_i(t) = '\phi' \wedge S_i(t) = B] = \Pr[S_i(t-1) = G]\cdot\alpha + \Pr[S_i(t-1) = B]\cdot(1-\beta)$$
$$\Pr[l_i(t) = '\phi' \wedge S_i(t) = G] = \Pr[S_i(t-1) = G]\cdot(1-\alpha) + \Pr[S_i(t-1) = B]\cdot\beta$$

<u>Induction step:</u> Assuming correctness of both claims for *n-1*, we prove them for *n*. Using conditional probabilities we get:

$$\Pr[E_i(t,n) = (l_i(t),\cdots,l_i(t+n-1)) \wedge S_i(t+n-1) = G] =$$
$$= \Pr[E_i(t,n-1) = (l_i(t),\cdots,l_i(t+n-2)) \wedge S_i(t+n-2) = G]\cdot$$
$$(\Pr[L_i(t+n-1) = l_i(t+n-1)) \wedge S_i(t+n-1) = G \mid$$
$$E_i(t,n-1) = (l_i(t),\cdots,l_i(t+n-2)) \wedge S_i(t+n-2) = G] +$$
$$\Pr[L_i(t+n-1) = l_i(t+n-1)) \wedge S_i(t+n-1) = G \mid$$
$$E_i(t,n-1) = (l_i(t),\cdots,l_i(t+n-2)) \wedge S_i(t+n-2) = B])$$

Since $L_i(t+n-1)$ and $S_i(t+n-1)$ depend on $E_i(t,n-1)$ and $S_i(t+n-2)$ only through $S_i(t+n-2)$, then the conditional probability is equal to:

$$\Pr[L_i(t+n-1) = l_i(t+n-1)) \wedge S_i(t+n-1) = G \mid S_i(t+n-2) = G] =$$

$$= \begin{cases} (1-\alpha_i)\cdot(1-P_B) & \text{if } l_i(t+n-1)='1' \\ (1-\alpha_i)\cdot P_B & \text{if } l_i(t+n-1)='0' \\ (1-\alpha_i) & \text{if } l_i(t+n-1)='\phi' \end{cases}$$

$$(\Pr[L_i(t+n-1)=l_i(t+n-1))\wedge S_i(t+n-1)=G \mid S_i(t+n-2)=B]=$$

$$= \begin{cases} \beta_i\cdot(1-P_B) & \text{if } l_i(t+n-1)='1' \\ \beta_i\cdot P_B & \text{if } l_i(t+n-1)='0' \\ \beta_i & \text{if } l_i(t+n-1)='\phi' \end{cases}$$

$$(\Pr[L_i(t+n-1)=l_i(t+n-1))\wedge S_i(t+n-1)=B \mid S_i(t+n-2)=G]=$$

$$= \begin{cases} \alpha_i\cdot(1-P_B) & \text{if } l_i(t+n-1)='1' \\ \alpha_i\cdot P_B & \text{if } l_i(t+n-1)='0' \\ \alpha_i & \text{if } l_i(t+n-1)='\phi' \end{cases}$$

$$(\Pr[L_i(t+n-1)=l_i(t+n-1))\wedge S_i(t+n-1)=B \mid S_i(t+n-2)=B]=$$

$$= \begin{cases} (1-\alpha_i)\cdot(1-P_B) & \text{if } l_i(t+n-1)='1' \\ (1-\alpha_i)\cdot P_B & \text{if } l_i(t+n-1)='0' \\ (1-\alpha_i) & \text{if } l_i(t+n-1)='\phi' \end{cases}$$

Now from the inductive assumption we have:

$$\Pr[E_i(t,n-1)=(l_i(t),\cdots,l_i(t+n-2))\wedge S_i(t+n-2)=G]=\pi_i^T(t-1)\left(\prod_{j=0}^{n-2}\widehat{\mathrm{A}}_i^{l_i(t+j)}\right)[1]$$

$$\Pr[E_i(t,n-1)=(l_i(t),\cdots,l_i(t+n-2))\wedge S_i(t+n-2)=B]=\pi_i^T(t-1)\left(\prod_{j=0}^{n-2}\widehat{\mathrm{A}}_i^{l_i(t+j)}\right)[2]$$

Putting all these together, the two claims are proved for *n* (based on *n-1*). Using the induction we complete the proof of the two claims.
Finally, *theorem* 1 follows immediately from **(33)** and **(34)**.

אוניברסיטת תל-אביב
הפקולטה למדעים מדוייקים
ע״ש ריימונד ובברלי סאקלר
בית הספר למדעי המתמטיקה

פיזור חבילות ואיכות שירות של יישומי Voice over IP ברשתות IP

חיבור זה מוגש כחלק ממלוי הדרישות לקבלת התואר
״מוסמך למדעים״ M.Sc. באוניברסיטת תל-אביב
בית הספר למדעי המתמטיקה
החוג למדעי המחשב

ע״י

זלטוקרילוב חיים

העבודה הוכנה באוניברסיטת תל-אביב
בהדרכתו של פרופסור חנוך לוי

סיוון תשס״ב

פיזור חבילות ואיכות שירות של יישומי Voice over IP ברשתות IP

תקציר

רשתות הדור הבא, והנטייה כיום לעבור לרשתות מבוססות IP יהפוך רשתות אלו לתשתית העיקרית להעברת יישומים כגון קול ווידאו (Voice and Video Over IP). הנושא המרכזי שיש לטפל בו לצורך הצלחת המעבר לרשתות מבוססות IP הוא מתן איכות השירות הנדרשת ליישומים אלו. הבעיה העיקרית בתחום זה נובעת מכך שרשתות IP לא ניבנו בבסיסן עם הבטחת איכות שירות כמו רשתות ATM למשל ("best effort").

בעבודה זו אנו בוחנים את ההשפעה של פיזור חבילות מעל מסלולים מקבילים על יישומי VoIP. פיזור חבילות ברשתות IP הוא מנגנון שבו חבילות של יישום מפוזרות בין מסלולים מקבילים, המובילים מהמקור ליעד, בהתאם לשיטת פיזור מוגדרת. פיזור החבילות יכול להיות ממומש על ידי שימוש במנגנון ניתוב המקור (source routing) ברשתות IP על ידי המקור או על ידי צמתים אחרים ברשת כמו למשל צמתים המממשים חלוקת עומס (load balancers) או רשתות וירטואליות של חברות מסחריות כגון Akamai אשר משתמשות בארכיטקטורה של תחנות קצה למימוש חלוקת עומס וניצול טוב יותר של הרשת.

טכנולוגיות של פיזור מידע בין ערוצים משמשת ברשתות שונות לצרכים שונים. ברשתות רדיו CDMA, משתמשים בפיזור מידע מטעמי ביטחון (security) וכדי לחלק רעשים בין על ערוצים בין משתמשי קצה להשגת ניצולת סטטיסטית טובה יותר של הרשת. כך למשל אם מספר ערוצים מתוך N רועשים, הרעש מחולק בין המשתמשים כך שכל אחד סופג חלק מהרעש ותחת ההנחה שהערוצים החדשים טובים מספיק ניצולת הרשת עולה. רעיון דומה של פיזור חבילות ברשתות IP מוצע להוריד התפרצויות (bursts) של מידע ובכך לחלק עומס בין ערוצים ב-[9][33]. רעיון נוסף המוצג ב-[17] מציע להשתמש בפיזור העתקים של חבילות למימוש מנגנון הגנה מפני איבודי חבילות כמנגנון טוב יותר מ-*FEC* (Forward Error Correction) ליישומי VoIP. פיזור חבילות ממומש בפועל כיום ברשתות IP באמצעות מחלקי עומס ברשת.

ישנם הרבה גורמים המשפיעים על איכות הקול ביישומי VoIP. ניתן לחלק גורמים אלו לשתי קבוצות, המנגנונים הטכנולוגיים וביצועי הרשת. המנגנונים הטכנולוגיים כוללים גורמים כמו סוג ה-codec (למשל G.711, G.729) והיכולות שלו, מנגנוני *PLC* (Packet Loss Concealment) ו-*FEC* (Forward Error Correction) וכו'. ביצועי הרשת נמדדים לרוב בשלושה מדדים: איבוד חבילות, השהיית חבילות ושונות השהית החבילות. מובן שככל שמדדים אלו גדלים האיכות צפויה לרדת. עם זאת ההשהיה המקובלת עבור יישומים דו-כיוונים (כגון שיחת טלפון) מוגבלת מסיבות פיזיות לסדרי גודל של 200-250 מילי-שניות. אי לכך בצורה גסה ניתן לתרגם את השהיית החבילות לאיבודן כאשר אלו עוברות את הסף, מאחר והיישום אינו יכול להשתמש בחבילות מאחרות. אנו נתמקד באיבודי החבילות כפי שרואה אותם היישום ללא הבחנה ממקור איבוד החבילה (בין אם זה איבוד חבילה אמיתי או השהייתה מעבר למותר).

מחקרים של יישומי VoIP מראים כי חוסר שביעות הרצון מאיכות השיחה עולה עם התגברות התפרצויות של איבוד חבילות. איבוד חבילות ממוצע, כפי שהוצג במספר מחקרים, אינו מאפיין מספיק לצורך אפיון השפעת הרשת על יישומי ה-VoIP. לצורך הערכה יותר מדויקת של השפעת הרשת יש לקחת בחשבון גם את מאפיין ההתפרצות (loss burstiness) של איבודי החבילות והמיקום של ההתפרצויות (recency) בשיחה. כאשר מביאים בחשבון את הגורמים האמורים יחד עם המנגנונים הטכניים של ה-codec ניתן לשערך בצורה טובה את איכות השיחה על פי ה-E-model (ראה [6][7][10]). התכונה של ירידת איכות השיחה עם התגברות ההתפרצויות באה לידי ביטוי גם במודל זה. כאשר מתבוננים במנגנונים כמו *FEC* או *PLC* תכונה זו היא מובנת מאליה. שני המנגנונים הללו באים לפצות על אובדן חבילות על ידי שימוש במידע ישן (*FEC* משתמש במידע על החבילות הישנות שמוצמד לחבילות חדשות יותר, ו-*PLC* מתבסס על אפנון המידע מהמקטעים האחרונים שהתקבלו), מכאן ברור שאובדן חבילות בהתפרצויות פוגע באפקטיביות של מנגנונים אלו במיוחד. אינטואיטיבית, גם האוזן האנושית מתנהגות בצורה דומה, בעוד שאנו בד"כ נתעלם מרעשים אקראיים, רעשים ארוכים יחסית מפריעים לשמיעה. מטעמים אלו אנו טוענים שבמקרים רבים יש להשתמש במדד של "קצב שגיאות נראות" (*NLR*– Noticeable Loss Rate) [18], כמדד מתואם לחישוב איכות הקול. המטריקה של *NLR* סופרת איבודי חבילות "קרובים" ומתעלמת מאיבודי חבילות הרחוקים אחד מהשני. עם זאת אין אנו טוענים כי מטריקה זו מספיקה לתפוס את השפעת הרשת על היישום, אלה שבהתבסס על [6][7][10][32], מטריקה זו מתואמת היטב עם איכות הקול (מובן שככל שה-*NLR* קטן יותר האיכות תהיה טובה יותר). מסיבות אלו נתמקד בעבודה זו ב-*NLR* הנצפה על ידי יישום ה-VoIP.

באשר מתבוננים על כל הנאמר עד כה נראה כי פיזור חבילות יכל להיות לשמש לשיפור איכות הקול בעיקר בגלל העובדה כי פיזור חבילות מקטין את ההתפרצויות של הרעשים (איבודי חבילות במקרה של רשתות IP). אנו בוחנים את נכונות טענה זו על ידי חישוב והשוואה של ה-*NLR* הנצפה על ידי שיחות תוך שימוש בשיטות פיזור שונות והשוואתן לשליחת חבילות רגילה ברשתות IP, דהיינו ללא פיזור חבילות.

האנליזה בעבודה זו מבוססת על ההנחה שאיבודי החבילות הנצפים על מסלולים ברשת הם חסרי זיכרון (Bernoulli loss model) או איבודי חבילות תחת מודל של התפרצויות הנקרא: Gilbert loss model. כאשר בוחנים שיטות פיזור שונות תחת איבודי חבילות חסר זיכרון, אנו מראים כי פיזור חבילות מקטין בממוצע את ה-*NLR*. תחת מודל Gilbert המסקנות דומות אם כי ישנם מקרים בהם פיזור חבילות עלול להרע את המצב, כתלות במאפייני הרשת. למרות שהתוצאות מראות כי פיזור חבילות אכן יכול לשפר את האיכות של יישומי VoIP, יש לציין כי פיזור חבילות כשיטה, עלול לגרום לתופעות רשת בעתיות כגון חבילות שמגיעות בניגוד לסדר השליחה ויש לבחון היטב את כדאיות המנגנון במידה ורוצים לממשו ככלי לשיפור האיכות.

מבנה העבודה הוא כדלהלן: בפרק 2 אנו מציגים את המודל של ה-*NLR* והנחות הייסוד בבסיס המודל של העבודה. לאחר מכן אנו פונים לניתוח מתמטי של שיטות לפיזור חבילות תחת מודל של איבודי חבילות חסר זיכרון (פרק 3) ותחת המודל של איבוד חבילות בהתפרצויות, מודל Gilbert, בפרק 4. עבור שני המודלים של איבודי חבילות אנו מחשבים

תחילה את ה-*NLR* הנצפה על ידי שיחה המועברת על גבי מסלול יחיד, ללא שימוש בפיזור חבילות. לאחר מכן אנו מחשבים את ה-*NLR* בסביבה של ריבוי מסלולים מקבילים ובוחנים שתי שיטות פיזור חבילות עיקריות: 1. פיזור חבילות אקראי, בו נבחר מסלול בצורה אקראית מבין אוסף המסלולים האפשריים על פי הסתברות שונה לכל מסלול. 2. פיזור חבילות דטרמיניסטי, שבו המסלול הנבחר עליו תשלח חבילה נבחר בצורה דטרמיניסטית על פי שיטה הנקבעת מראש. שיטה טיפוסית לפיזור כזה תהיה פיזור מחזורי. לאחר שנציג את הניתוח המתמטי אנו נשווה מספרית שיטות פיזור אלו ונציג מסקנות.