# Improved Quality for Conversational VoIP using Path Diversity

*Qipeng Gong, Peter Kabal*

Electrical & Computer Engineering, McGill University
Montreal,Quebec,Canada H3A 2A7
qi.gong@mail.mcgill.ca peter.kabal@mcgill.ca

## Abstract

In Voice-over-IP, the quality of interactive conversation is important to users. Quality-based playout buffering seeks an optimum balance between delay and loss. However, such a scheme still suffers when packet losses are bursty. Path diversity can alleviate the effect of losses and improve perceived quality by providing redundancy. In this paper, a new scheme is proposed which evaluates the performance of both paths. We consider three different path diversity schemes. The playout scheduling algorithms are designed based on conversational quality including both calling quality and interactivity. The simulation results show the efficacy of our algorithms in correcting for losses (isolated and burst) and improving perceived conversational quality.

## 1. Introduction

In VoIP, factors associated with perceived call quality are delay, delay jitter, and packet loss. All of these factors stem from the "best effort" nature of IP networks. Missing packets include both network losses (packets that never arrive) and late packets (resulting in buffer underflow). Network losses can be caused by: link failure, heavy network load, and/or packet collisions. Under heavy loading conditions, packets in the queues in routers may need to be dropped. Late packets occur when packets arrive at the receiver after they are scheduled to be played out. These late packets are of concern for the design of playout buffer at the receiver side. Even though, a long buffer reduces the number of late packets, the conversational delay is increased, with a consequent impact on interactivity. If packet losses are bursty, degradation on perceived quality is more than that caused by isolated losses [1]. Therefore, it is desirable to improve perceived quality by reducing packet losses without adding further delay.

Packet loss concealment (PLC) algorithms are used to fill in the missing speech frames. However, PLC techniques are not very effective at concealing long bursts of packet losses. Moreover, in some cases, to save transmission bandwidth, multiple speech frames are packetized together with the effect that a single loss may result in a burst loss of speech frames [2]. Most PLC schemes are designed to gradually mute the output when consecutive frames are erased.

Forward error correction (FEC) [3] can be used to mitigate the impact of packet losses by sending redundant information. *Signal Processing FEC* (SP-FEC) piggybacks redundant information onto subsequent packets. Most FEC schemes require additional delay to use redundant information. In [1], we proposed a new SP-FEC scheme without additional delay, and the results showed that conversational quality was improved with reduced packet losses.

FEC schemes can only protect against a small number of missing packets. A path diversity scheme is an alternative which uses multiple paths (here we consider two paths). Redundant information is sent on a second path. If the loss and delay characteristics of the two paths are uncorrelated, path diversity schemes are robust to burst losses. The information on a second path can be full redundancy or partial redundancy. In a full redundancy scheme, packets are duplicated. In a partial redundancy scheme, only important packets (those which have a significant effect on perceived quality if lost), are duplicated and sent on a second path. In this way, network loading is reduced. However, importance detection at the sender's side is typically complex, and for some applications, the increase in bit rate for fully duplication is preferred to an increase in complexity. Bit rate can be reduced by using a lower rate-compression to encode the redundancy packets on a second path. For example, use G.711 (64 kb/s) on the "default" path (path 1), and GSM coding (13 kb/s) or G.729 coding (8 kb/s) on the second path (path 2).

Either IP source routing or relay approaches can be used to implement path diversity schemes. With IP source routing, special configurations are required for all nodes that a packet might visit on route to its destination [4]. Relay approaches use relays placed at a number of strategic nodes to forward a packet to its destination. In this paper, we consider the latter approach.

In the work of Ghanassi [4], path diversity was used to improve the perceived quality for a E-model-based playout algorithm. In [4], both fully redundancy and partial redundancy schemes are considered. In a full redundancy scheme, the receiver selects the first arriving packet to reconstruct the speech signal and the minimum of the current delays for the two paths is used to estimate the playout delay for following packets. Although Ghanassi's method guarantees acceptable conversational delays, delays can be further reduced by two steps as in [5]. The scheme for reducing conversational delays is described in Section 3.2. With conversational interactivity in mind, we propose quality-based playout designs using three different path diversity schemes. We also propose a new path diversity scheme using the quality evaluation for both paths. The path diversity schemes are discussed in Section 3.2. The results show that conversational quality of adaptive playout buffering is improved by the use of path diversity.

The contributions of this paper are three-fold:

1. A new estimation of jitter buffer size based on diverse paths (*Scheme 3*)

2. Quality-based playout algorithms with path diversity schemes, reducing conversational delays

3. Investigation of the robustness to burst losses for adaptive playout buffering, algorithms using path diversity, and FEC based algorithms.

## 2. Forward Error Correction (FEC) vs. Path Diversity

In Signal Processing FEC, redundant information is added to subsequent packets. To use this redundant information, the decoder must implement a delay. Since a jitter buffer is already present at the receiver, there need be no additional delays if SP-FEC is integrated with the jitter protection algorithm. In [1], we proposed an adaptive SP-FEC scheme without additional delay as Fig. 1. At a sender side, $m$ previous voiced packets are piggybacked, but piggybacking is stopped whenever "hangover" [1]is detected (VAD/DTX from the G.729 [6]). The value of $m$ is determined by the jitter buffer size at the receiver side and is communicated to the sender by a RTCP packet. The benefit of reconstructing missing packets declines with increasing length of burstiness.

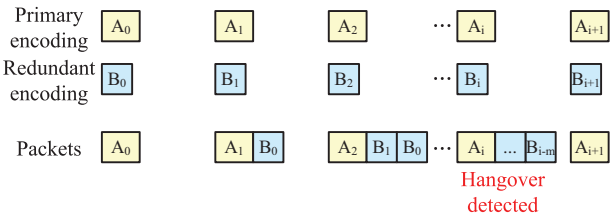


Figure 1: SP-FEC scheme in [1]

Path diversity allows for redundancy packets to be carried by a second path. Additional delays can be avoided with path diversity. When used with quality-based playout algorithm, the packets carrying redundancy can also used to estimate the jitter buffer size. Figure 2 illustrates the scheduling process using path diversity with the adaptive buffering in Section 3.2. Note that the packets at the start of a talkspurt are stretched (shown as being longer) to build up the buffer delay, and packets are compressed when "hangover" is detected. The overall bit rate can be lowered by either sending only important packets or using a lower rate-compression encoder (with lower attendant quality and higher complexity).

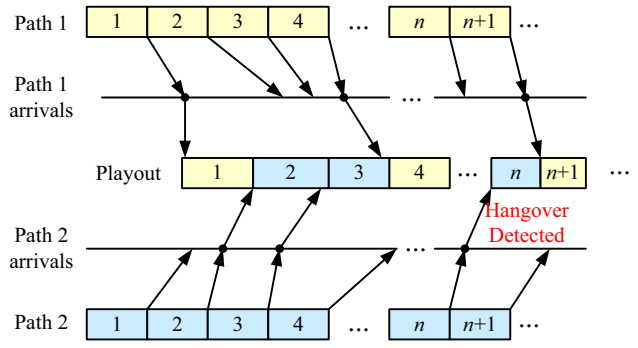


Figure 2: Path diversity scheme

## 3. Playout Scheduling Algorithm using Path Diversity

For VoIP, conversational delay should be kept as low as possible to allow for interactivity. In this paper, our design of playout buffering is based on the optimization criterion proposed in our previous work [5], which takes into account both voice quality and conversational delay. Path diversity offers us the opportunity to lower the delay and losses even further.

[1]a unvoiced packet sent as a voiced packet to avoid speech clipping

### 3.1. E-Model-based Conversational Quality Maximization

The conversational quality measurement is

$$Q_c = R + g(D_c), \tag{1}$$

where $R$ is ITU-T E-model R factor, $D_c$ is the conversational delay which is calculated using the method proposed in our previous work [5]. Although $g(\cdot)$ is unknown so far, the relation between $Q_c$ and $D_c$ is: $Q_c$ goes down when $D_c$ goes up, and vice versa. Hence, maximization of $Q_c$ is equal to maximizing the $R$ factor and/or minimizing $D_c$.

According to [7], the R factor can be written as

$$R = 93.2 - I_e - I_d, \tag{2}$$

where $I_d$ is the delay impairment factor, and $I_e$ is the equipment impairment factor. Denoting $d$ as the end-to-end delay, the factor $I_d$ can be derived by a simplified fitting process from [8],

$$I_d = 0.024d + 0.11(d - 177.3)\,H(d - 177.3), \tag{3}$$

where

$$H(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0. \end{cases}$$

The equipment impairment factor is codec dependent. For G.711 with PLC, it can be approximated as [4]

$$I_e = I_{ec} + I_\rho = 0 + 7\ln(1 + 50\rho), \tag{4}$$

where $I_{ec}$ is the impairment caused by encoder (0 for G.711), and $\rho$ (in percentage) is the packet loss including network loss and late packets.

According to [5], Eq. (2) can be written as

$$\begin{aligned} R &= 93.2 - (I_d + I_\rho) \\ &= 93.2 - \big[0.024d + 0.11(d - 177.3)\,H(d - 177.3) \\ &\qquad + 7\ln\big(1 + 50(\rho_n + \rho(d))\big)\big], \end{aligned} \tag{5}$$

where $\rho_n$ is the network loss and $\rho(d)$ is the loss causing buffer underflow for a delay $d$ (in $ms$). The term $\rho(d)$ can be calculated as

$$\begin{aligned} \rho(d) &= (100 - \rho_n)P(X > d) \\ &= (100 - \rho_n)(1 - F(d)), \end{aligned} \tag{6}$$

where $F(d)$ is the cumulative distribution function (CDF) of the delay. In this paper, $F(d)$ is calculated as a function of playout delay using the histogram of the most recent $w$ packet delays. In our simulation, $w = 1000$ packets.

### 3.2. Playout Scheduling Algorithms using Path Diversity

Human speech consists of talk-spurts interspersed with silence. Packet losses during talk-spurts degrade the perceived quality dramatically, while losses during silence period cause almost no effect on the perceived quality. Therefore, many playout scheduling algorithms tune a jitter buffer at the beginning of each talk-spurt. Compared with continuously updating approaches, a per-talkspurt approach implements smoother playout speech.

In this paper, Eq. (1) is used as optimization criterion for the design of playout scheduling. As in [5], the conversational delay $D_c$ is reduced by two steps. First, the first packet of a talk-spurt is stretched and played out as soon as it arrives. This stretching process increases the buffer depth. Second, at the end of a talkspurt, compress the voiced packets in the jitter buffer whenever the "hangover" packet is detected. Our previous work in [5] and [1] shows the efficiency of reducing conversational delays by these steps.

The following operations are performed at the receiver side:

- During a silence period, comfort noise is played out every 10 ms, whether a Silence Insertion Description (SID) packet arrives or not. The jitter buffer size $d_{\rm jb}$ is zero. Information about packet losses and transmission delay are saved. SID packets are used to update the comfort noise parameters.
- When the first packet of the first talk-spurt arrives, packet WSOLA (PWSOLA) (see [9]) is applied to stretch the decoded speech before it is played out. The jitter buffer size increases by $(\alpha - 1) \times T_F$ ($\alpha$ is the stretch factor, and $T_F$ is the payload length of a packet). The $d_{\rm jb}$ parameter is estimated based on previously stored packet delay information (window size is 1000 packets).
- When the estimated $d_{\rm jb}$ is achieved, the decoded speech is not stretched any further. The depth of jitter buffer keeps the steady-state value $d_{\rm jb}$ and sets $\alpha = 1$.
- At the end of a conversation turn, when the hangover is detected, PWSOLA is applied to compress the decoded speech before it is played out. The jitter buffer size decreases by $(1-\alpha) \times T_F$. Compression stops when jitter depth is decreased to zero. It is possible for "hangover" to happen in the middle of the talk-spurt, for example, during the silence gap within a word. In this case, we stretch the subsequent voiced packet as if it were the beginning of the talk-spurt. A noticeable change in the silence gap can be avoided [10].

At sender side, the speech signal is encoded and packetized before is sent both to path 1 and path 2. For simplicity, we use G.711 as encoder for both paths. At receiver side, redundancy packets on path 2 are used to recover missing packets from path 1, that is, if a packet is lost or arrives after it is scheduled to play out, the corresponding redundancy packet received from path 2 is used to reconstruct the speech. The delay information of redundancy packets is also used to estimate $d_{\rm jb}$.

In this paper, we introduce 3 schemes to estimate $d_{\rm jb}$ using path diversity. The term $p_{d1}$ is the end-to-end delays on path 1, and $p_{d2}$ is the end-to-end delays on path 2

**Scheme 1** In this scheme, the first arrived packet on both paths is used. If the performances of the two paths are highly correlated, there is no significant gain in improving quality [4]. The following is performed:

*For every packet, use* $d = \min(p_{d1}, p_{d2})$ *to update the delay window.*

*At the beginning of a talkspurt, find the* $d_{opt}$ *which maximizes* $R \rightarrow d_{\rm jb}$.

**Scheme 2** Path 2 is used only for reducing the packet loss on path 1, i.e., the delays on path 2 are used only when the corresponding packets are lost on path 1.

*For every packet, use*

$$d = \begin{cases} p_{d1}, & \text{packet on path 1 arrives} \\ p_{d2}, & \text{packet on path 1 is lost.} \end{cases}$$

*to update the delay window.*

*At the beginning of a talkspurt, find* $d_{opt}$ *which maximizes* $R \rightarrow d_{\rm jb}$.

**Scheme 3** This a new scheme proposed. We evaluate the performance for both paths, and choose the minimum $d_{\rm jb}$ for the current talkspurt.

*For every packet, use*

$$d^{(1)} = \begin{cases} p_{d1}, & \text{packet on path 1 arrives} \\ p_{d2}, & \text{packet on path 1 is lost.} \end{cases}$$

*and*

$$d^{(2)} = \begin{cases} p_{d2}, & \text{packet on path 2 arrives} \\ p_{d1}, & \text{packet on path 2 is lost.} \end{cases}$$

*to update the delay windows for path 1 and path 2 respectively.*

*At the beginning of a talkspurt, search* $d_{opt}^{(1)}$ *and* $d_{opt}^{(2)}$ *which maximize* $R$ *in Eq.* (5)*,* $d_{opt} = \min(d_{opt}^{(1)}, d_{opt}^{(2)}) \rightarrow d_{\rm jb}$.

*Scheme 1* gives the smallest $d_{\rm jb}$ in the case that two paths are uncorrelated. The $d_{\rm jb}$ obtained from *Scheme 2* is slightly different from that estimated on a single path (path 1), because $p_{d2}$ is used when a packet is lost on path 1, which changes CDF in Eq. (6). Indeed, *Scheme 2* can achieve high performance when $p_{d1} > p_{d2}$, and hence gives the highest $d_{\rm jb}$. In *Scheme 3*, the $d_{\rm jb}$ is between those from *Scheme 1* and *Scheme 2*, which keeps the jitter buffer reasonable short.

### 3.3. Robustness to Burst Losses

Packet loss is a main factor influencing perceived quality. Burst losses degrade perceived quality. In this section, we investigate the robustness of our algorithms to burst losses. A 3-minute speech file is packetized to 8177 packets. To evaluate algorithms, the network channel is modelled from our internet trace file from Canada to China (see [5]), and a 2-state Gilbert Model is superposed to generate network losses. The transition probabilities are set such that the network loss is 5% and that an expected burst length ($E[BL]$) is achieved. The $E[BL]$ is varied from 1 packet to 19 packets (20 ms per packet). When $E[BL] = 1\times$packet, the packet loss is random, with no burst loss. The two paths for path diversity are simulated by randomly choosing two uncorrelated segments from the trace file.

Figure 3 shows the performances of different playout algorithms to different $E[BL]$: *Algorithm 1*, *Algorithm 2* and *Algorithm 3* are the adaptive algorithm described in Section 3.2 with *Scheme 1*, *Scheme 2*, and *Scheme 3*, respectively. *FEC* is the algorithm proposed in [1], which uses adaptive SP-FEC to send redundancy. *Adaptive* is the adaptive algorithm in [5] without redundancy transmission. Perceived quality is calculated objectively using PESQ [11], whose output is PESQ_MOS score. Note that PESQ only measures quality ignoring delays. *FEC* algorithm fails to improve perceived quality when $E[BL] \geq 7$ while the three algorithms with path diversity keep the PESQ_MOS score hight when $E[BL]$ increases. Therefore, path diversity schemes are more robust to the burst losses, and can improve perceived quality than the other two algorithms. Among the three algorithms using path diversity, *Algorithm 2* achieves highest performance with highest PESQ_MOS because *Scheme 2* gives the highest $d_{\rm jb}$ for a talkspurt, which reduces the impact of buffer underflow (late packets), and *Algorithm 3* performs between the other two algorithms.

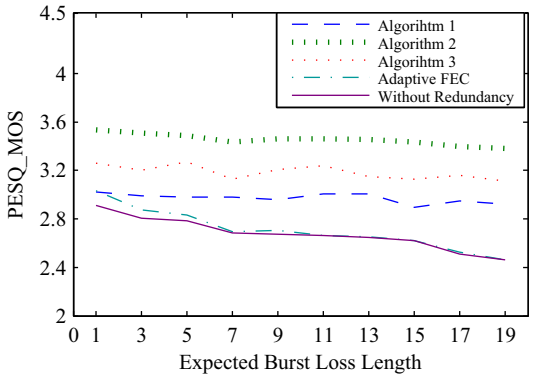


Figure 3: Performance Comparison with different E[BL].

## 4. Results

To simulate the transmission over the internet, we use three delay trace files: *Trace 1* was collected in January, 2009 Canada to China (see [5] for details), and *Trace 2* and *Trace 3* are from [8] between the UK and China. The network packet loss is superposed by a 2-state Gilbert Model. The network loss rate is 5% and the expect of network burst loss length is 5 packets. The conversation used in our simulation is from the recording of a real dialog (with noisy background), which consists of conversation turns, in an "ask-response" pattern. The five algorithms in Section 3.3 are applied and compared, and the results are shown in Table 1.

Table 1: Performance Comparison of Playout Buffering Algorithms for Internet Traces

| Trace | Buffering algorithms | Conversational delay (ms) | PESQ-MOS | PLR (%) |
|---|---|---|---|---|
| 1 | Algorithm 1 | 316.3 | 3.18 | 7.7 |
| | Algorithm 2 | 334.4 | 3.52 | 2.3 |
| | Algorithm 3 | 334.0 | 3.52 | 2.3 |
| | FEC | 334.4 | 2.75 | 9.0 |
| | Adaptive | 334.4 | 2.70 | 9.6 |
| 2 | Algorithm 1 | 155.9 | 3.94 | 0.3 |
| | Algorithm 2 | 192.6 | 3.94 | 0.3 |
| | Algorithm 3 | 188.5 | 3.94 | 0.3 |
| | FEC | 192.6 | 2.94 | 6.8 |
| | Adaptive | 192.6 | 2.76 | 7.4 |
| 3 | Algorithm 1 | 327.7 | 2.78 | 7.4 |
| | Algorithm 2 | 329.0 | 3.02 | 5.1 |
| | Algorithm 3 | 329.0 | 3.02 | 5.1 |
| | FEC | 329.0 | 1.65 | 24.0 |
| | Adaptive | 329.0 | 1.57 | 24.6 |

According to Table 1, the algorithms with path diversity improve the perceived quality without increasing conversational delay. The results also show that path diversity schemes work better than the SP-FEC scheme for reducing packet loss. *Algorithm 1* achieves the lowest conversational delay because the jitter buffer size is shorter than *Algorithm 2* and *Algorithm 3*. For the same reason, the late packets are more likely to be dropped and accordingly PLR (packet loss rate) for speech packets is higher than other two path diversity algorithms. *Algorithm 2* performs same as *Algorithm 3* in *Trace 1* and *Trace 3* because $d_{\mathrm{jb}}$ estimated from path 1 is smaller than that from path 2. In the case that path 2 gives smaller $d_{\mathrm{jb}}$, as in *Trace 2*, *Algorithm 2* achieves better conversational quality with smaller conversational delay.

## 5. Conclusions

In VoIP, conversational quality includes perceived quality and interactivity which is measured by conversational delays. Perceived quality can be improved by redundancy information. A new path diversity scheme is proposed to be used in E-model-based playout scheduling, which is based on the performance on both paths. Without increasing conversational delays, our quality-based algorithms with different path diversity schemes improve the conversational quality for quality-based adaptive buffering. The results also show that path diversity schemes achieve higher performances than SP-FEC.

## 6. References

[1] Q. Gong and P. Kabal, "Quality-Based Playout Buffering with FEC for Conversational VoIP," in *Proc. Interspeech* (Makuhari, Japan), pp. 2402–2405, Sept. 2010.

[2] J. Benesty, M. M. Sondhi, and Y. Huang (Eds.), *Springer Handbook of Speech Processing*, Springer-Verlag, 2008.

[3] J-C. Bolot, S. Fosse Parisis, and D. Towsley, "Adaptive FEC-based Error Control for Internet Telephony," in *Proc. IEEE Infocom* (New York, NY), pp. 1453–1460, March 1999.

[4] M. Ghanassi and P. Kabal, "Optimizing Voice-over-IP Speech Quality Using Path Diversity," in *Proc. IEEE Workshop Multimedia Signal Processing* (Victoria, BC), pp. 155–160, Oct. 2006.

[5] Q. Gong and P. Kabal, "A New Optimum Jitter Protection for Conversational IP," in *IEEE Int. Conf. Wireless Communications & Signal Processing* (Nanjing, China), pp. 1–5, Nov. 2009.

[6] ITU-T, *A Silence Compression Scheme for G.729 Optimized for Terminals Conforming to Recommendation V.70*, ITU-T Rec. G.729 Annex B, Jan. 2007.

[7] ITU-T, *The E-Model, a Computational Model for Use in Transmission Planning*, ITU-T Rec. G.107, March 2003.

[8] L. Sun and E. Ifeachor, "New Models for Perceived Voice Quality Prediction and their Applications in Playout Buffer Optimization for VoIP Networks," in *Proc. IEEE ICC* (Paris, France), pp. 1478–1483, June 2004.

[9] Y. -J. Liang, N. Farber, and B. Girod, "Adaptive Playout Scheduling using Time-Scale Modification in Packet Voice Communications," in *IEEE ICASSP* (Salt Lake City, UT), pp. 1445–1448, June 2001.

[10] M. Lee, J. McGowan, and M. C. Recchione, "Enabling Wireless VoIP," *Bell Labs Technical J.*, vol. 11, pp. 201–215, Nov. 2007.

[11] ITU-T, *Perceptual Evaluation of Speech Quality (PESQ), an Objective Method for End-to-end Speech Quality Assessment of Narrowband Telephone Networks and Speech Codecs*, ITU-T Rec. P.862, Nov. 2005.