Advanced Time Shrinking Using a Drop Classifier Based on Codec Features

Jochen Issing[1], *Nikolaus Färber*[2], *and Reinhard German* [1]

[1]Computer Networks and Communication Systems, Dept. of Computer Science,
University of Erlangen, Germany
[2]Fraunhofer Institut Integrierte Schaltungen, Erlangen

jochen.issing@cs.fau.de, nikolaus.faerber@iis.fraunhofer.de,
reinhard.german@fau.de

10.21437/Interspeech.2015-170

Abstract

We present an integrated approach of full-band audio time scale modification for Voice over IP communication. The concept is based on a low complexity adaptive playout method that uses frame dropping and audio concealment for time shrinking and stretching, respectively. The existing version of this method is improved using a classifier that assists in choosing which audio frames can be dropped with the least subjective impact on audio quality. To maintain low complexity, we exclusively use audio signal features that are available in the audio codec. The classification of audio frames improves audio quality of the existing method without classification by 0.5 Mean Opinion Score points while requiring significantly less computational complexity by a factor of ca 10^4.

Index Terms: Voice over IP, human conversation

1. Introduction

When a digital audio frame travels from the sender in RTP-packets [1] through the internet to the receiver, it passes many different devices and sub systems. Because subsequent packets can take different network paths, or might get queued in some devices due to cross traffic, etc. the propagation delay of audio packets can vary notably. This variation is also known as jitter [1] and is an inherent characteristic of IP-based networks. Due to the continuous nature of audio samples, we have to compensate the jitter in the receiver to maintain continuous playout of audio frames, which otherwise leads to audible drop outs. This jitter compensation is what we call Adaptive Playout (AP) [2] and is a key mechanism used in VoIP clients to compensate variable network delay. In the case of audio communication, AP is a very important component on the way to a high quality audio Communication System (CS). AP needs to balance a difficult trade-off between late loss and communication delay. If an audio frame arrives later than its actual playout time, it is considered as late loss and discarded, resulting in audible artifacts. With a larger receiver buffer, this late loss occurs less likely but communication suffers from additional delay. Besides that, jitter is not a stationary process but has the tendency to drift over time and often is interrupted by network delay spikes (i.e., short term buffering effects). To compensate and adapt to such effects, the receiving VoIP client needs to stretch and shrink the audio buffer dynamically to adapt to current network conditions.

2. Related Work

We introduced a Flexible Playout Adaptation [3] which exploits the overlap-add structure of transform based codecs for low complexity time scale modifications and discussed its relation to existing adaptive playout mechanisms [4] [5] [6] [7]. It uses audio frame dropping to shrink and concealment to stretch the audio signal as shown in Figure 1

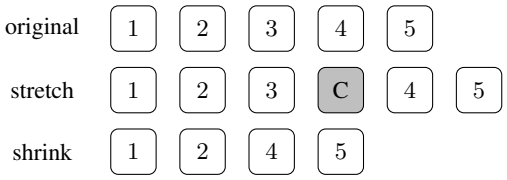

Figure 1: Conceptual example of stretching and shrinking audio using packet dropping or concealment insertion.

The method was compared to another industry standard method and performed well for most signal classes up to 4 % time shrinking [8] while requiring no additional complexity.

3. Motivation

The main goal of this paper is to improve the perceived audio quality of the shrinking method introduced in [3]. Therefore, we refine the time shrinking algorithm by adding a frame classifier which predicts the audibility of a dropped audio frame. The classification is based on specific audio signal features which are already available in the audio codec. The classifier shall facilitate the selection of the frames to be dropped in the receiver while retaining the best possible audio quality and low complexity.

4. Design of a Classifier for Frame Dropping

To train the classifier, we manually classified single frames in terms of their drop-ability. To cover a broad range of signals with an emphasis on speech, we chose nine different audio stimuli we also refer to as training signals. Six of those were speech items and three contained music. The speech items comprise two clean speech items and four with background noise including street, car, and office noise.

Each of those items is encoded using a low delay version of Unified Speech and Audio Coding (USAC) [9] (which we

Copyright © 2015 ISCA

also address as LD-USAC) and saved as bitstream files. Each bitstream i with N_i frames was then decoded and saved in *drop audio files* consisting of 100 frames (e.g., 50 frames before and 50 frames after the dropped frame). Thereby, each bitstream results in $N_i - 100$ drop audio files. Furthermore, a decoded version without frame drop serves as a reference for the classification process.

At least three listeners classified the audibility of each frame by comparing the corresponding drop audio file with the reference version. The classification is realized using four ordinal categories as shown in Table 1 and we calculate the overall result using the median. In total, 1769 frames have been classified

Table 1: Classification keys with description

Key	Value	Description
D	0	Not audible (Dropable)
A	1	Slightly audible
AA	2	Audible
AAA	3	Annoying

using the described method.

4.1. Audio Signal Features

The feature set upon which the classification tree is created is based on the features which are already available in the encoding process. The features include the signal energy, the Zero-Crossing-Rate, which counts the times the audio signal changes its sign, the Pearson product-moment correlation coefficient (PCC), the Pitch as well as a measure for the periodicity.

Some of the features were normalized based on the frame length. We estimate the pitch of a frame of audio samples x_i with window size N using the autocorrelation function [10] [11].

$$\rho(i) = \sum_{j=0}^{N-1} x_j \bar{x}_{j+i}, \tag{1}$$

$$\lambda = \arg\max(\,\rho(i)\,),\ 2\,ms \leq i \leq 20\,ms. \tag{2}$$

We normalize the pitch lag so that it represents the relative amount of phase shift.

$$L'_\lambda = 1 - \left| \frac{L_\lambda}{2\lambda} - 1 \right| \tag{3}$$

Using the pitch's wave length from Equation 2 and the correlation function from Equation 1 we determine the periodicity by

$$P = \frac{1}{4}\left(\rho(\lambda) + \rho(2\lambda) - \rho(\frac{\lambda}{2}) - \rho(\frac{3\lambda}{2})\right). \tag{4}$$

4.2. Classification Based on Decision Trees

We build the classification tree by using the approach of [12] since it provides a classification tree build method that is robust against overfitting and selection bias towards covariates with many splits. It is available as the **party** package in GNU/R [13]. The authors implement recursive binary partitioning using the theory of permutation tests developed by [14]. The algorithm features non-parametric class of regression trees and is applicable to all kinds of regression problems including nominal, ordinal, numeric, censored, and multivariate response variables. The splitting criteria is based on two-sample linear statistics rather than an impurity measure and therefore does not require additional pruning of the resulting tree.

We have chosen to use all covariates, i.e., signal features, as the input to the classification process. The response is the reduced manual classification conducted previously and consists of each dropable frame (D) as is and each audible frame (A,AA,AAA) simplified to audible (A). The resulting classifier will therefore predict each frame to be either dropable or audible.

4.2.1. Low Delay USAC classifier

We build the classifier for low delay version of USAC (LD-USAC) by using all classified frames in the classifier creation process. The LD-USAC codec has been submitted as a candidate for the Enhanced Voice Service (EVS) [15] [16] [17] [18] [19] as a next generation joined audio and speech codec for the Long Term Evolution (LTE) standard.

Table 2 shows the confusion matrix for the classification tree. In total, 77.3 % of all frames have been classified correctly.

	A	D	correct	wrong
A	253	239	51.4%	48.6%
D	162	1114	87.3%	12.7%
total	415	1353	77.3%	22.7%

Table 2: Confusion matrix of low delay USAC classifier. Original classes are given per row while classifier results are presented in columns. In total, 76.5% of all frames are classified as dropable.

When considering misclassified frames, it's important to note that an original *A/AA/AAA* frame which was incorrectly classified as *D* results in audible artefacts whereas an original *D* frame which was incorrectly classified as *A/AA/AAA* only reduces the maximum shrinking rate.

To reduce the amount of wrongly classified audible frames, we therefore add a weight to each observation of our classification training data. The weights will give observations with stronger impact on audio quality more weight than others and thereby reduce the amount of misclassified audible frames. We derive the weights w_i from the classification level values v_i by

$$w_i = v_i \cdot f_w + 1 \tag{5}$$

with $v_i \in \{0, 1, 2, 3\}$ and the constant factor f_w to adjust the weight levels manually. Even though there is a risk to overspecify the classification tree with too high weight factors and thereby increase the number of nodes notably, it is for the benefit of reduced misclassified audible frames.

Table 3 shows the effect of different weight factors when creating the classification tree. The misclassification rate for originally audible frames decreases significantly with increasing f_w until a factor of 2. However, the number of frames classified as dropable also decreases from 76.5 % to 49.6 %. This limits the maximum shrink rate accordingly. Apart from that, the total number of nodes in the non-reduced classification trees raises to more than double the size. A weighting factor of 3 shows not much difference apart from a seriously increasing number of nodes. We interpret this as a sign of saturation and we have seen this effect growing for higher weight factors. Therefore, we will use a weighting factor of 2 for our low delay USAC classifier.

f_w	A misclass. rate	max shrink rate	#nodes
0	48.6%	76.5%	21
1	29.5%	64%	37
2	14.4%	47.9%	45
3	13.4%	47.5%	55

Table 3: Effect of different weights on relative dropable misclassifications, maximum shrink rate and total number of nodes per classification tree.

Table 4 shows the confusion matrix for the final low delay USAC classification tree with weight factor 2.

	A	D	correct	wrong
A	421	71	85.6%	14.4%
D	501	775	60.7%	39.3%
total	922	846	67.6%	32.4%

Table 4: Confusion matrix of low delay USAC classifier with weight factor 2. Original classes are given per row while classifier results are presented in columns. In total, 846 (47.9%) of all frames are classified as dropable. Also note that only 14.4% of all audible frames are incorrectly classified as droppable, assuring improved audio quality.

4.3. Complexity Estimation

To estimate the complexity of the low delay USAC dropping classifier, we use the so-called BASOP (Basic Operator Libraries) instrumentation specification of the ITU, which is incorporated in [20]. The instrumentation results in the two complexity parameters WMOPS (Weighted Million Operations per Second) and ROM (Read-only Memory) usage, which estimate computational complexity in terms of speed and memory usage.

The complexity of an if-statement is given as 4 complexity weights and 2 memory weights by [20]. Based on these values, we can make an complexity estimation by simple tree analyzation. The minimum complexity is given by the least amount of if-clauses necessary to classify a frame and is in this example 2. The worst case scenario is given by all frames classified by 7 if-clauses in total. Apart from the extreme values, we can estimate the expected average complexity using the classification results from tree creation. We therefore extract each fitting node (n_j), i.e., the node that finally classifies the frame, from the observations. The if-rate r_{if} which determines the classifiers WMOPS and ROM complexity measures is given by

$$N_{if} = \sum_{i=1}^{29} nifs(n_{j,i}) \tag{6}$$

$$r_f = 50 \tag{7}$$

$$r_{if} = \frac{N_{if}}{N_f \cdot r_f}, \tag{8}$$

with $nifs$ counting the number of if-clauses for each fitting node, N_{if} being the total number of if-clauses for all classified frames, r_f being the frame rate in 1/s.

Table 5 summarizes minimum, maximum and expected complexity values for the low delay USAC classification tree. Note that complexity is in general estimated using WMOS, which are million operations per second, where our results are orders of magnitudes below.

Table 5: Complexity estimation of low delay USAC classification tree

	WOPS	WORDS
min	400	14
expected	838	14
max	1400	14

4.4. Subjective Evaluation

To verify the functionality and to assess the quality of the classifier we have to conduct a verification listening tests.

4.4.1. Verification Test Design

The ITU P.800 specification [21] recommends different methods for listening only audio quality assessment. With Absolute Category Rating (ACR), the test subjects are exposed to a single stimulus, i.e., a shrunken file or coded content they are only allowed to listen to once. After listening, the test subjects provide their rating on a discrete Mean Opinion Score (MOS) scale, i.e., numbers from 1 to 5 representing the opinions *bad*, *poor*, *fair*, *good*, and *excellent*.

We have chosen 6 pairs of items from the McGill speech database [22] as well as two music pieces for our verification test. The speech items were around 5 s long, while the music items lasted for about 10 s, to allow the subjects capturing the tune of the music as well as the artifacts. To 5 of the 6 speech items we added background noise. The background noise comprised separately recorded music, office, street, car, as well as babble noise. Each noise signal was normalized to -51 dBov according to P.56 [23], except for the office noise, which was normalized to -46 dBov. All other items have been normalized to -26 dBov, as required by the P.800 specification.

We tested the following shrinking algorithms and MNRUs (Modulated Noise Reference Units):

- USAC2 (classifier)
- USAC2 regular dropping (e.g. every 10th frame)
- USAC2 random dropping (using uniform random numbers)
- WSOLA [24]
- MNRU 34 dB
- MNRU 16 dB

To cover a broad range of shrinking rates, we created test files with 4 %, 10 %, and 20 % shrinking. In total, the listening test comprises 144 files. We decided to split the test files into two groups, to reduce the time for each test subject to about 15 min. We used a P.800 test software that automatically scrambles the audio files and splits them into groups.

4.4.2. Verification Test Results

The test subjects listened to the test audio signals using headphones and a custom listening test software guided through the test process. We tested 30 subjects in 15 test groups. The results contain 2160 ratings in total with 15 ratings per test item.

Figure 2 shows the results of the P.800 listening test over all rates. The original (Ref) and both MNRU items contain 120 ratings while all shrinking algorithms consist of 360 ratings each. WSOLA is included as a high quality reference. As can be seen from Figure 2 when comparing LDUSAC with LDUSAC reg, the overall improvement of the classifier across all rates is almost half a MOS point. Even though the classifier does not reach high profile algorithms like WSOLA, which performs one MOS point better, we have improved the dropping algorithm at negligible performance cost.

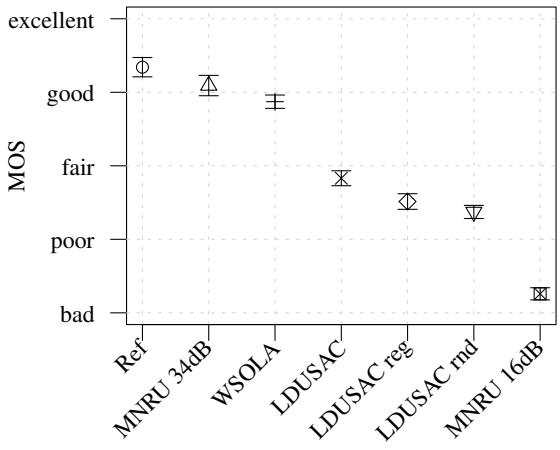

Figure 2: Test results of the P.800 listening test to verify functionality of the classifiers for low delay USAC (LDUSAC) over all rates (4 %, 10 %, 20 %) and all items. The results are given as mean values with 95 % confidence intervals according to the t-test. The reference (Ref), the 34 dB MNRU, and WSOLA showing good audio quality, while the MNRU 16 dB item has been rated bad in average by the test subjects. In between, we have the group with the classifiers and the regular and random dropping schemes. The classifier improves quality by almost half a MOS point and thereby comes close to fair audio quality.

We now focus on speech and therefore exclude all music items from the analysis. Apart from that, we reduced the results to contain only dropping rates of 4 % and 10 % as the effect of classification diminishes towards higher dropping rates.

Figure 3 shows the summarized results of all algorithms for shrinking rates of 4 % and 10 % and speech only. LDUSAC with random dropping shows bad audio quality, even compared to regular LDUSAC dropping. Listening into the items turned out that the dropped frames are often clustered closely together with random dropping and therefore amplified the audio degradation. In contrast to that, regular dropping has a constant distance between two frame drops and results in higher average audio quality as not every frame drop is equally audible.

Even though this does not reach the quality of a high-profile algorithm, the improvement is remarkable especially without increasing the complexity of the algorithm. We can state that the classifier works well at shrinking rates up to 10 %, especially for speech, and provides about half an MOS point better quality than simple frame dropping.

Note that the classifier has been built with a reduced feature set, i.e., features already available in the encoding process. A free selection of signal features should make it possible to reach higher quality levels. Besides that, adaptation is not a permanent operational mode but only necessary in adaptation periods, e.g. when network conditions change.

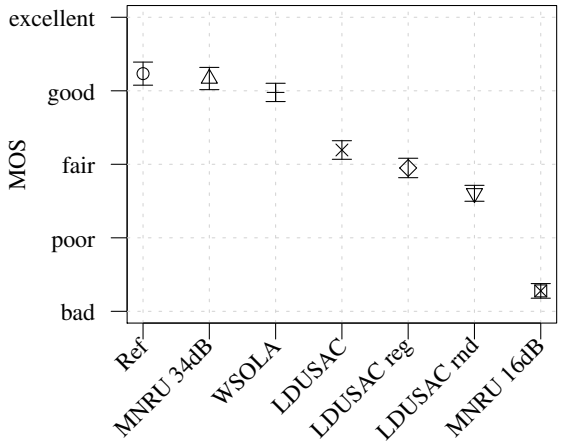

Figure 3: Test results of the P.800 listening test to verify functionality of the classifiers for low delay USAC (LDUSAC) and reduced to speech items and rates of 4 % and 10 %. The results are given as mean values in a 95 % confidence interval according to the t-test.

As a final note, we compare the classifier's complexity to WSOLA. Table 6 lists the estimation results from G.191 instrumentation for the applied implementations regarding average WMOPS and ROM usage (WORDS):

Table 6: Complexity of time shrinking algorithms

algorithm	WMOPS	WORDS
WSOLA	$5.5 - 6.6$	295
Classifier	$838 \cdot 10^{-6}$	14

Even though the classifiers cannot compete with high profile algorithms like WSOLA in terms of audio quality, the improvement in audio quality from poor to fair for 10 % shrinking is remarkable, especially as the computational complexity reduction is about 1×10^4. The classifiers are therefore interesting candidates for low power devices, e.g. smartphones, and for low shrinking rates up to 4 %, where the additional complexity of WSOLA has only little benefit.

5. Conclusions

We have further improved the previously published transform-codec-based time shrinking mechanism using a frame classifier. The frame classifier predicts the audibility of allegedly dropped frames based on signal features which are already available in the USAC codec. Compared to the previous design, we have increased the possible shrink rate from 4 % to 10 % for roughly the same quality. Overall, we could improve the time shrinking audio quality by another 0.5 MOS points at approximately zero complexity cost.

6. References

[1] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," Tech. Rep., Jul. 2003.

[2] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE Network*, vol. 12, no. 5, pp. 40–48, 1998.

[3] J. Issing, F. Dressler, N. Färber, and S. Reuschl, "Flexible Playout Adaptation for Low Delay AAC RTP Communication," in *The Third International Conference on COMmunication Systems and NETworkS (COMSNETS 2011)*, Bangalore, India, Jan. 2011.

[4] Y. J. Liang, N. Faerber, and B. Girod, "Adaptive playout scheduling and loss concealment for voice communication over IP networks," in *Adaptive playout scheduling and loss concealment for voice communication over IP networks*, 2003, pp. 532–543.

[5] S. B. Moon, J. Kurose, and D. Towsley, "Packet audio playout delay adjustment: performance bounds and algorithms," in *Packet audio playout delay adjustment: performance bounds and algorithms*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1998, pp. 17–28.

[6] R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne, "Adaptive Playout Mechanisms for Packetized Audio Applications in Wide-Area Networks," in *INFOCOM '94. Networking for Global Communications*, Toronto, Canada, Jun. 1994, pp. 680–688.

[7] H. Schulzrinne, "Voice communication across the Internet: a network voice terminal," Tech. Rep., 1992.

[8] J. Issing and N. Färber, "Subjective Audio Quality Impairments of Playout Adaptation with AAC RTP Communication," in *International Communications Quality and Reliability (CQR) Workshop*, Naples, Florida, USA, May 2011.

[9] M. Neuendorf, P. Gournay, M. Multrus, J. Lecomte, B. Bessette, R. Geiger, S. Bayer, G. Fuchs, J. Hilpert, N. Rettelbach, R. Salami, G. Schuller, R. Lefebvre, and B. Grill, "Unified speech and audio coding scheme for high quality at low bitrates," in *International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, Taipei, Taiwan, Apr. 2009, pp. 1–4.

[10] X. Huang, A. Acero, and H. W. Hon, *Spoken Language Processing: A guide to Theory, Algorithm, and System Development*. Prentice Hall, Jan. 2001.

[11] D. Zwillinger, *CRC Standard Mathematical Tables and Formulae*. Chapman and Hall/CRC, 2012.

[12] T. Hothorn, K. Hornik, and A. Zeileis, "Unbiased Recursive Partitioning: A Conditional Inference Framework," *Journal of Computational and Graphical Statistics*, vol. 15, no. 3, pp. 651–674, Sep. 2006.

[13] R. C. Team, *R: A Language and Environment for Statistical Computing*, 2012.

[14] H. Strasser and C. Weber, "On the Asymptotic Theory of Permutation Statistics," *Mathematical Methods of Statistics*, vol. 8, pp. 220–250, 1999.

[15] "Universal Mobile Telecommunications System (UMTS); LTE; Codec for Enhanced Voice Services (EVS); Performance characterization."

[16] T. A. L. Ericsson, F. IIS, H. T. C. Ltd, N. Corporation, NTT, N. DOCOMO, INC, ORANGE, P. Corporation, Q. Incorporated, S. E. Co, Ltd, VoiceAge, and Z. Corporation, "High Level Technical Description of the Jointly Developed EVS Candidate Codec," Seoul, South Korea, Tech. Rep. 77th TSG-SA4 Meeting, Jan. 2014.

[17] S. Bruhn, H. Pobloth, M. Schnell, B. Grill, J. Gibbs, L. Miao, K. Järvinen, L. Laaksonen, N. Harada, N. Naka, S. Ragot, S. Proust, T. Sanda, I. Varga, C. Greer, M. Jelinek, M. Xie, and P. Usai, "Standardization of the new EVS Codec," in *40th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, Australia, Apr. 2015.

[18] D. Martin, M. Multrus, V. Eksler, V. Malenovsky, E. Norvell, H. Pobloth, L. Miao, Z. Wang, L. Laaksonen, A. Vasilache, Y. Kamamoto, K. Kikuiri, S. Ragot, J. Faure, H. Ehara, V. Rajendran, V. Atti, H. Sung, E. Oh, H. Yuan, and C. Zhu, "Overview of the EVS Codec Architecture," in *40th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015.

[19] V. Atti, D. Sinder, S. Subasingha, V. Rajendran, D. Dewasurendra, V. Chebiyyam, I. Varga, V. Krishnan, B. Schubert, J. Lecomte, X. Zhang, and L. Miao, "Improved error resilience for VOLTE and VOIP with 3GPP EVS channel aware coding," in *40th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015.

[20] "ITU-T Recommendation G191: Software tools for speech and audio coding standardization," ITU-T Telecommunication Standardization Sector, Mar. 2010.

[21] "ITU-T Recommendation P.800: Methods for subjective determination of transmission quality," ITU-T Telecommunication Standardization Sector.

[22] P. Kabal, "TSP Speech Database," Telecommunications & Signal Processing Laboratory Department of Electrical & Computer Engineering, McGill University, Tech. Rep., Sep. 2002.

[23] "ITU-T Recommendation P.56: Objective measurement of active speech level," ITU-T Telecommunication Standardization Sector, Dec. 2011.

[24] W. Verhelst and M. Roelands, "An overlap-add technique based on waveform similarity (WSOLA) for high quality time-scale modification of speech," in *Multiple values selected*. Minneapolis, Minnesota, USA: IEEE, Apr. 1993, pp. 554–557.