Audio Packet Loss Concealment in a Combined MDCT-MDST Domain

Hadas Ofir, David Malah, *Fellow, IEEE*, and Israel Cohen, *Senior Member, IEEE*

***Abstract*—Audio streaming applications have become very popular in recent years, owing to their low cost and convenience. However, during network congestions, data packets are often delayed or discarded, creating an annoying gap in the streamed media. This letter presents a new approach to audio packet loss concealment designed for MPEG-Audio streaming applications. In a previous work, we introduced a receiver-based concealment algorithm based on applying the gapped-data amplitude and phase estimation (GAPES) interpolation algorithm in the discrete short-time Fourier transform (DSTFT) complex domain and obtained better results compared to past methods. The current approach applies the same algorithm on a different complex domain, formed from combining the modified discrete cosine transform (MDCT) domain as its real part and the modified discrete sine transform (MDST) domain as the imaginary part. The new approach significantly reduces the complexity demands while maintaining similar high-quality results.**

***Index Terms*—Audio coding, discrete cosine and sine transforms, gapped-data interpolation, packet loss concealment.**

I. Introduction

AUDIO transmission over the internet is called *audio streaming* since the data flows in a digital stream from the server to the client, ready to be heard in real time, without having to download it all before use. One of its main problems is packet loss. Since internet delivery does not guarantee quality of service, data packets are often delayed or discarded during network congestions. Missing packets create a gap in the streamed audio, and the client's audio player has nothing to play. This is an interpolation problem, where the missing signal is reconstructed in a perceptual sense, so that a human listener does not notice the disturbance.

Many packet loss recovery techniques exist in the literature, e.g., [1] and [2], mostly designed for speech applications, and few for music signals. One of the reasons is that music signals are sampled at a higher rate than speech signals (44.1 kHz versus 8 kHz); hence, a loss of even a single audio packet, which usually corresponds to 20–30 ms of audio [3], creates a wide gap in samples (~ 1000). Time-domain interpolation methods dealing with such a wide gap yield very poor results and hence are appropriate only for low loss rates ($< 10\%$). Previous works on MPEG-audio packet loss concealment include simple solutions such as packet repetition, as suggested in the MP3 standard [3], and more sophisticated solutions, designed for higher loss rates, that are applied in the spectral domain. Such methods include the statistical interpolation (SI) algorithm [4], which applies interpolation in the compressed (MDCT) domain, by treating each time-trajectory of the coefficients for a given frequency bin as a separate signal with missing samples. Other algorithms in this category use MAPES-CM [5] or GAPES [6] interpolation algorithms in the DSTFT domain [7], [8]. The above solutions are listed in order of increasing complexity but also with increasing quality, where the last algorithm, which we refer to as GAPES-in-DSTFT, provides the best results. Its superiority is mainly attributable to using a complex spectral domain, where the signal representation is less fluctuating, whereas in the MDCT domain, the coefficients typically show rapid sign changes from frame to frame in each frequency bin [9]. Interpolation in the DSTFT domain requires, however, conversions from MDCT to DSTFT, and vice versa. Such conversions add complexity to the decoder, and even though efficient conversions were developed and used in [7], the overall complexity is still quite high.

In this letter, we present a new algorithm, which applies GAPES in a complex spectral domain formed by considering the MDCT and MDST coefficients as the real and imaginary components, respectively. The MDCT coefficients are available in the compressed domain, and the MDST coefficients are calculated directly from the MDCT coefficients by a simple procedure that requires seven times **less** the number of multiplications as compared to the DSTFT conversion. The results show that the new algorithm provides at least the same level of concealment quality as the algorithm in [7], at a lower complexity. The remainder of this letter is organized as follows: Section II presents more considerations in choosing the concealment domain and describes the process of calculating the MDST coefficients. Section III describes the new concealment algorithm, and Section IV presents the results of subjective quality tests. Finally, Section V concludes this letter.

II. Concealment Domain

In the case of audio coding, two concealment domains come immediately to mind: the time domain and the compressed domain. MPEG-audio coders [10], [11] compress the signal in the MDCT domain. Specifically, the MP3 encoder divides the signal into 50% overlapping segments of 576 samples each. Then, each segment is converted to the MDCT domain by using one of four possible window functions, resulting in 576 MDCT coefficients. The different windows enable representation of different segments at different time-frequency resolutions, according to their short- or long-term characteristics. Each MP3 packet contains two such segments. In the decoder, the signal is restored using

Manuscript received April 12, 2007; revised June 21, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Patrick A. Naylor.

The authors are with the Department of Electrical Engineering, Technion-Israel Institute of Technology, Technion City, Haifa 32000, Israel (e-mail: hadaso@tx.technion.ac.il; malah@ee.technion.ac.il; icohen@ee.technion.ac.il).

Digital Object Identifier 10.1109/LSP.2007.904711

an overlap-and-add (OLA) procedure on the output of the inverse MDCT [12]. This means that a loss of a single MP3 packet affects the reconstruction of three segments, i.e., 1728 samples. However, in the MDCT domain, a lost packet is interpreted as a small gap of two consecutive coefficients at each frequency bin. Since a smaller gap is easier to handle, it was suggested in [4] to conceal the data loss in the MDCT domain, rather than in the time domain.

However, working in the MDCT domain has its limitations: First, as mentioned before, the rapid sign changes that are typical to the MDCT coefficients make them difficult to interpolate. Working in a complex domain, which has a less fluctuating representation of the signal, should provide better interpolation results. For this reason, we previously [7] interpolated the missing data in the DSTFT domain. Another alternative, proposed here, is to use the MDST coefficients along with the MDCT coefficients to create a complex representation of the signal. A second problem is that since different window types have different frequency resolutions, the MDCT coefficients of two consecutive frames at a certain frequency bin might represent different frequency resolutions. In this case, it does not make sense to interpolate the data separately in each frequency bin based only on the correlation along the time axis. Applying 2-D interpolation, which exploits the frequency bins inter-correlation, may overcome this limitation, however, at the expense of higher complexity, and hence, it is not pursued further here. Another possible solution, applied in this work, is to recalculate the MDCT coefficients for a single window type. This is done by converting them back to the time domain and then applying the MDCT again with a fixed window.

A. Calculating MDST From MDCT

The MDST coefficients are calculated from the available information, i.e., the MDCT coefficients. Both are real-valued transforms, turning $2N$ time samples into N spectral coefficients, defined by the following:

MDCT:

$$C_k^{(p)} = \sum_{n=0}^{2N-1} x_n^{(p)} \cdot h_n^{(p)} \cdot \cos\left[\frac{\pi}{N}\left(n+\frac{N+1}{2}\right)\left(k+\frac{1}{2}\right)\right] \tag{1}$$

MDST:

$$S_k^{(p)} = \sum_{n=0}^{2N-1} x_n^{(p)} \cdot h_n^{(p)} \cdot \sin\left[\frac{\pi}{N}\left(n+\frac{N+1}{2}\right)\left(k+\frac{1}{2}\right)\right] \tag{2}$$

where $0 \le k \le N-1$, $x_n^{(p)}$ is the original signal segment, $h_n^{(p)}$ is the window function of length $2N$, and $p \in Z$ is the index of the time segment.

In order to calculate the MDST coefficients, each time-domain segment ($2N$ samples) is reconstructed by using an inverse-MDCT transform followed by an overlap-and-add process (same as in the conversion to DSTFT, see [13, Appendix B]). Then, an MDST transform is applied to each segment. After some algebraic manipulations, a more efficient conversion is given by

$$\begin{aligned} S_k^{(p)}\Big|_{1\le k\le N-2} &= \left(C_{k-1}^{(p)} - C_{k+1}^{(p)}\right)\cdot e_1 \\ &\quad + (-1)^k \sum_{m=0}^{N-1} C_m^{(p+1)} \cdot e_3[k,m] \\ &\quad + \sum_{m=0}^{N-1} (-1)^m\, C_m^{(p-1)} \cdot e_2[k,m] \end{aligned} \tag{3}$$

$$\begin{aligned} S_{N-1}^{(p)} &= \left(C_{N-1}^{(p)} + C_{N-2}^{(p)}\right)\cdot e_1 \\ &\quad + (-1)^{N-1} \sum_{m=0}^{N-1} C_m^{(p+1)} \cdot e_3[N-1,m] \\ &\quad + \sum_{m=0}^{N-1} (-1)^m\, C_m^{(p-1)} \cdot e_2[N-1,m] \end{aligned} \tag{4}$$

$$\begin{aligned} S_0^{(p)} &= \left(C_0^{(p)} - C_1^{(p)}\right)\cdot e_1 + \sum_{m=0}^{N-1} C_m^{(p+1)} \cdot e_3[0,m] \\ &\quad + \sum_{m=0}^{N-1} (-1)^m\, C_m^{(p-1)} \cdot e_2[0,m] \end{aligned} \tag{5}$$

where e_1 and $e_{2-3}[k,m]$ for $0 \le k,m \le N-1$ are defined as follows:

$$e_1 = \frac{1}{N}\sum_{n=0}^{N-1}\left[\left(h_n^{(p)}\right)^2 - \left(h_{n+N}^{(p)}\right)^2\right]\sin\left[\frac{\pi}{N}\left(n+\frac{N+1}{2}\right)\right] \tag{6}$$

$$\begin{aligned} e_2[k,m] &= \\ &\frac{1}{N}\sum_{n=0}^{N-1} h_{n+N}^{(p-1)} \cdot h_n^{(p)} \left\{\cos\left[\frac{\pi}{N}\left(n+\frac{N+1}{2}\right)(k+m+1)\right]\right. \\ &\left. - \cos\left[\frac{\pi}{N}\left(n+\frac{N+1}{2}\right)(k-m)\right]\right\} \end{aligned} \tag{7}$$

$$\begin{aligned} e_3[k,m] &= \\ &\frac{1}{N}\sum_{n=0}^{N-1} h_n^{(p+1)} \\ &\cdot h_{n+N}^{(p)} \left\{\cos\left[\frac{\pi}{N}\left(n+\frac{N+1}{2}\right)(k+m+1)\right]\right. \\ &\left. + \cos\left[\frac{\pi}{N}\left(n+\frac{N+1}{2}\right)(k-m)\right]\right\}. \end{aligned} \tag{8}$$

It is worthwhile noting that in order to maintain constant frequency resolution in the concealment domain, the MDST is calculated using a fixed window of type "long." Also, for the same reason, an MDCT frame that uses a non-long window type is converted to "long" by reconstructing its corresponding time-domain samples ($2N$) and applying an MDCT again, this time using a long window. In this case, it is simpler to calculate the corresponding MDST coefficients by applying the transform directly to the reconstructed samples.

Regarding the complexity of this calculation, the constant e_1 and the expressions $e_2[k,m]$ and $e_3[k,m]$ can be calculated offline, stored, and read from a lookup-table (LUT). In our application, since all the MDCT frames are converted, so they use the same window type, there is no dependence on index p, and hence, the LUT contains only $2N^2+1$ real-values. Assuming an LUT is employed, the calculation of all N MDST coefficients requires $N(2N+1)$ multiplications and the same number of additions. Comparing these values with the conversion from MDCT to the DSTFT domain and vice versa, which requires $14N^2$ multiplications and $28N^2$ additions [8], clearly shows a significant improvement. Also, it is important to note that the new algorithm does not require backward conversion (i.e., MDST to MDCT), as is required in the DSTFT-based algorithm.

III. Concealment Algorithm

The concealment block, located in the decoder, is similar to the one presented in our previous works [7], [8]. In short, every new MP3 packet is decoded up to the MDCT level (i.e., de-quantized), resulting in two MDCT frames. The P most recent MDCT frames and their corresponding window types are stored in a buffer. If delay is allowed, the buffer may contain future (yet unplayed) packets. Thus, the parameter P would affect the delay before play out starts.

If a packet is lost, the MDCT frames corresponding to that packet are set to zero, and the type of their unknown window functions are determined so that they comply with the neighboring frames. Each frame, when its turn arrives, is copied from the buffer and decoded into waveform samples. In case of a lost frame, its MDCT coefficients are reconstructed using the concealment algorithm before continuing with the decoding process. The algorithm has the ability to conceal several lost frames simultaneously, in what we refer to as a *concealment session*: lost frames that are located close to each other are usually concealed together, while distant losses are concealed in separate sessions.

The concealment algorithm itself is illustrated in Fig. 1. It starts with a buffer of P consecutive MDCT frames, some of which are missing. First, the corresponding MDST coefficients are calculated for each MDCT frame in the buffer, based on that frame and its two closest neighbors, using the procedure described in Section II-A. As mentioned above, frames that use a non-long window type are converted to "long." It is important to note that if one of the non-long frame, or one of its closest neighbors is missing, then the resulting "new" MDCT and MDST frames contain time-domain aliasing that could not be removed and hence are considered as "corrupted" (see Fig. 1).

The upper-left square in Fig. 1 starts the iterative part of the algorithm: The time-trajectories of the MDCT and MDST coefficients for each frequency bin are treated as separate complex signals with missing samples, and a single iteration of the GAPES algorithm is applied to each such signal. The GAPES algorithm [6] reconstructs missing data, assuming it has the same spectral content as the available data. The algorithm was applied here in the same way as in [7]. After applying this to all the bins, we have an estimated version of the missing MDCT frames and their corresponding MDST frames. The last step in the iteration is the following: If a missing frame has one or two neighbors that

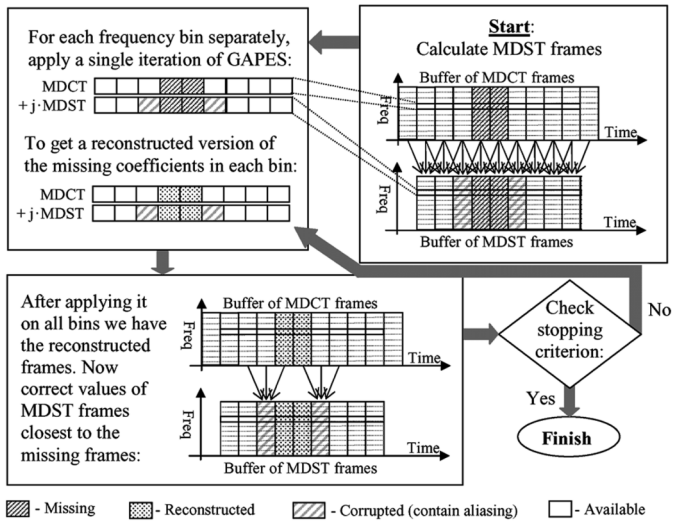

Fig. 1. Diagram of the concealment algorithm.

TABLE I
Examined Files

No.	File Name	Nature of Music
1	Beatles17.wav	Pop music
2	Piano10.wav	A single piano
3	Bream1.wav	Guitar with violins
4	Jazz6.wav	Jazz music
5	Flute2.wav	Flute and piano

are "available" (i.e., not missing), their corresponding MDST frames are recalculated based on the most recent estimation of the lost MDCT frame, in order to reduce the effect of unremoved aliasing that these MDST frames contain. The process above is iterated until the difference between consecutive reconstructions becomes small. Then, the reconstructed MDCT frames are used instead of the missing ones, and the MP3 decoding process continues.

IV. Results

The performance of the new algorithm was evaluated using subjective listening tests, since available objective measures, even perceptual ones, did not reflect the sensation created in a human listener [13]. The tests were carried out by informal listening, using 12 inexperienced listeners with good hearing. Each listener was asked to compare pairs of audio files, where the packet losses in each file in the pair were concealed by a different method, and to decide which of the two he, or she, prefers. Since a previous report [7] already showed the advantage of the GAPES-based algorithms over earlier reported methods, the new algorithm was compared to just two algorithms, referred to as GAPES-in-DSTFT and GAPES-in-MDCT [7]. In the latter case, non-long window frames are first converted to long window frames, as done in the new algorithm, and then GAPES is applied directly on the real-valued MDCT coefficients.

Table I specifies the files used in the tests. All are stereo signals, 15–17 s long, sampled at 44.1 kHz and coded by the LAME MP3 encoder at a bit-rate of 128 kb/s per channel. Since for 10% loss rate the concealed losses are practically unnoticeable, the test

TABLE II
COMPARATIVE TEST RESULTS OF GAPES-IN-MDCT/MDST VERSUS GAPES-IN-DSTFT.
THE NUMBERS INDICATE HOW MANY LISTENERS VOTED IN FAVOR OF EACH METHOD

	File No. 1		File No. 2		File No. 3		File No. 4		File No. 5	
Loss Rate	GAPES-in-MDCT/MDST	GAPES-in-DSTFT	GAPES-in-MDCT/MDST	GAPES-in-DSTFT	GAPES-in-MDCT/MDST	GAPES-in-DSTFT	GAPES-in-MDCT/MDST	GAPES-in-DSTFT	GAPES-in-MDCT/MDST	GAPES-in-DSTFT
20%	5	**7**	4	**8**	**11**	1	2	**10**	**7**	5
30%	**9**	3	5	**7**	**10**	2	**6**	**6**	**12**	0

TABLE III
COMPARATIVE TEST RESULTS OF GAPES-IN-MDCT/MDST VERSUS GAPES-IN-MDCT.
THE NUMBERS INDICATE HOW MANY LISTENERS VOTED IN FAVOR OF EACH METHOD

	File No. 1		File No. 2		File No. 3		File No. 4		File No. 5	
Loss Rate	GAPES-in-MDCT/MDST	GAPES-in-MDCT	GAPES-in-MDCT/MDST	GAPES-in-MDCT	GAPES-in-MDCT/MDST	GAPES-in-MDCT	GAPES-in-MDCT/MDST	GAPES-in-MDCT	GAPES-in-MDCT/MDST	GAPES-in-MDCT
20%	**8**	4	**10**	2	**8**	4	**10**	2	**7**	5
30%	**10**	2	**12**	0	**10**	2	**11**	1	**12**	0

focused on high loss rates, such as 20% and 30%, with random loss patterns. Table II shows the listeners' votes, compared to GAPES-in-DSTFT. On average, for 20% loss, there is almost a tie: 48.3% of the listeners voted in favor of the new algorithm, while 51.6% preferred otherwise. For 30% loss, however, most of the listeners (70%) preferred the new algorithm. So it is safe to say that the performance of the new algorithm, in terms of quality of the resulting signal, is about the same as the DSTFT algorithm, and even exceeds it for high loss rates. When compared to the GAPES-in-MDCT algorithm, the voting results, presented in Table III, confirm that the complex MDCT-MDST domain is more suitable for interpolation than the MDCT-domain: 71.6% of the listeners voted for the proposed algorithm for 20% loss rate, and 91.6% preferred it for 30% loss rate.

V. CONCLUSION

A new algorithm is introduced for packet loss concealment in a complex-domain formed by combining the MDCT and MDST representations. The new algorithm yields about the same quality of concealment results as the algorithm in the DSTFT domain, which was ranked best until now, and even outperforms it for high loss rates. Furthermore, the proposed algorithm is significantly more efficient than the latter, since the complexity requirements of using GAPES are much lower in a complex MDCT-MDST domain than DSTFT domain.

REFERENCES

[1] B. W. Wah, X. Su, and D. Lin, "A survey of error-concealment schemes for real-time audio and video transmissions over the internet," in *Proc. Int. Symp, Multimedia Software Engineering*, Taipei, Taiwan, R.O.C.,, Dec. 2000, pp. 17–24.

[2] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE Network.*, vol. 15, no. 5, pp. 40–48, Sep.-Oct. 1998.

[3] ISO/IEC 11172-3, "Coding of moving pictures and associated audio for digital storage media at up to 1.5 MBit/s-Part 3: Audio," ISO/IEC JTC 1/SC 29, 5/20/1993.

[4] S. Quackenbush and P. Driessen, "Error mitigation in MPEG-audio packet communication systems," in *Proc. 115th AES Conv.*, New York, Oct. 2003, pp. 1–11.

[5] Y. Wang, J. Li, and P. Stoica, "Two-dimensional nonparametric spectral analysis in the missing data case," in *Proc. 30th IEEE Int. Conf. Acoustics, Speech, Signal Processessing (ICASSP-2005)*, Philadelphia, PA, Mar. 2005, pp. 397–400.

[6] P. Stoica and E. G. Larsson, "Adaptive filter-bank approach to restoration and spectral analysis of gapped data," *Astronom. J. Amer. Astronom. Soc.*, pp. 2163–2173, 2000.

[7] H. Ofir and D. Malah, "Packet loss concealment for audio streaming based on the GAPES algorithm," in *Proc. 118th AES Conv.*, Barcelona, Spain, May 2005, pp. 1–19.

[8] H. Ofir and D. Malah, "Packet loss concealment for audio streaming based on the GAPES and MAPES algorithms," in *Proc. 24th IEEE Conv. Electrical and Electronics Engineers in Israel*, Eilat, Israel, Nov. 2006, pp. 280–284.

[9] J. M. Tribolet, "Frequency domain coding of speech," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 5, pp. 512–530, Oct. 1979.

[10] D. Pan, "A tutorial on MPEG/audio compression," *IEEE Multimedia*, vol. 2, no. 2, pp. 60–74, Summer 1995.

[11] K. Brandenburg, "MP3 and AAC explained," in *Proc. AES 17th Int. Conf. High Quality Audio Coding*, Signa, Italy, Sept. 1999, pp. 1–17.

[12] Y. Wang and M. Vilermo, "Modified discrete cosine transform-its implications for audio coding and error concealment," *AES J.*, vol. 51, no. 1/2, pp. 52–61, 2003.

[13] H. Ofir, "Packet loss concealment for audio streaming" Master's thesis, Technion-Israel Inst. Technol., Haifa, Israel, 2006. [Online]. Available: http://www-sipl.technion.ac.il/new/Research/Publications/Graduates/HadasOfir/HadasMScthesis.pdf.