Enhanced MPEG-4 Low Delay AAC - Low Bitrate High Quality Communication

Markus Schnell[1], Ralf Geiger[1], Markus Schmidt[1], Manuel Jander[1], Markus Multrus[1], Gerald Schuller[2], Jürgen Herre[1]
[1] *Fraunhofer IIS, Erlangen, Germany*

[2] *Fraunhofer IDMT, Ilmenau, Germany*

Correspondence should be addressed to Markus Schnell (`Markus.Schnell@iis.fraunhofer.de`)

ABSTRACT
The MPEG-4 Low Delay Advanced Audio Coding (AAC-LD) scheme has recently evolved into a popular algorithm for audio communication. It produces excellent audio quality at bitrates between 64 kbit/s and 48 kbit/s per channel. This paper introduces an enhancement to AAC-LD which reduces the bitrate demand by 25-33%. This is achieved by adding both a delay-optimized version of the Spectral Band Replication (SBR) tool and by utilizing a dedicated low delay filterbank. The introduced techniques maintain the high audio quality and offer an algorithmic delay low enough for use in two way communication systems. This paper describes the coder enhancements including a detailed discussion of algorithmic delay issues, a performance assessment and possible applications.

1. INTRODUCTION

MPEG-4 ER Advanced Audio Coding Low Delay (AAC-LD) [1] has recently enjoyed increasing adoption as a full bandwidth, high quality communication coder. Several manufacturers of advanced video- and teleconferencing systems incorporated this MPEG-4 audio codec into their products to guarantee their customers low latency communication with high fidelity.

In comparison, standard speech codecs, such as ITU-T G.729.1, usually work at lower bitrates but come with some limitations, including a focus on single-speaker speech material, unsatisfactory performance for music signals and a limited audio bandwidth. In order to obtain the audio quality of AAC-LD as well as the low bitrate demand of speech codecs, a further enhancement of the AAC-LD coding efficiency is desirable.

Currently, AAC-LD produces excellent audio quality at a bitrate range of 64 kbit/s to 48 kbit/s per channel. In this paper a new codec, called "Enhanced Low Delay AAC (AAC-ELD)", will be presented which extends the range of operation down to 24 kbit/s per channel. The codec is an extension of the AAC-LD scheme and utilizes additional coding tools.

The Spectral Band Replication (SBR) tool, well known from MPEG-4 HE-AAC (see Section 2.3), has proven to be an attractive enhancement to audio coders for low bitrate coding. The simple combination of AAC-LD and SBR would, however, result in a total algorithmic delay of 60 ms (see Section 4), thus rendering the codec unsuitable for communication applications. Generally, the system delay for interactive two way communication should not exceed 50 ms. The modifications necessary to keep the delay sufficiently low will be presented in this paper. The new technical components for AAC-ELD are:

- a modification of the SBR tool in order to minimize the system delay
- a replacement of the MDCT filterbank by a dedicated low delay filterbank in order to alleviate the remaining delay increase

As a result, the AAC-ELD coder exhibits a delay well within the acceptable range for bi-directional communication, and saves about 25-33% of bitrate compared to regular AAC-LD while maintaining the level of audio quality. The scheme described in this paper is currently the object of an ISO/IEC MPEG standardization process.

This paper is organized as follows. An overview of the underlying codecs is provided in Section 2. In Section 3 the new codec is presented followed by detailed discussion of the algorithmic delay in Section 4. Section 5 continues with a quality assessment of the new system. In Section 6 implementation aspects are disscussed and Section 7 presents a concise overview of potential applications.

2. BACKGROUND: MPEG-4 AAC LC, MPEG-4 ER AAC LD AND MPEG-4 HE-AAC

In this section, the MPEG-4 state-of-the-art general audio codecs that form the basis of the Enhanced Low Delay AAC coder are reviewed briefly.

2.1. MPEG-4 AAC LC

Designed as a successor to MPEG Layer-3 [2][3], AAC has quickly become the basis of MPEG Audio within MPEG-2/4. The standardization process of this general audio coder has been finalized in 1997 as MPEG-2 AAC [4] and updated in 1999 within MPEG-4 [1]. In particular AAC Low Complexity (AAC-LC), the low-complexity subset of AAC, is widely used in various application scenarios such as broadcasting, Internet download services etc. AAC-LC delivers good quality starting at 32 kbit/s per channel and approaches perceptual transparency from around 64 kbit/s/channel.

As Figure 1 shows, AAC is designed as a filterbank based audio coder. A time domain audio signal is transformed into the spectral domain using a Modified Discrete Cosine Transform (MDCT). The spectral components are scaled and quantized according to the requirements of a psychoacoustic model before they are entropy coded and multiplexed into a bitstream. For a more detailed description see [5].

2.2. MPEG-4 ER AAC LD

While AAC-LC provides high audio quality, its algorithmic delay of at least 55 ms (1024 samples per frame, 48 kHz) is clearly too high for bi-directional communication. Derived from AAC-LC, a low-delay general audio coder was introduced within MPEG-4 [1] as MPEG-4 ER AAC LD (AAC-LD) [6][7].

With a reduced transform size, a newly introduced low-overlap window and the deactivation of the block switching mechanism, the codec achieves an optimized algorithmic delay of down to 20 ms. Excellent audio quality can be reached starting from 48 kbit/s per channel.

2.3. MPEG-4 HE-AAC

The next milestone in MPEG-4 towards low bit rate coding was the introduction of SBR, a generic parametric coding tool for high frequencies. The combination of SBR and AAC-LC was standardized in 2003 in the MPEG-4 High-Efficency (HE-AAC) [8] and achieves FM quality at bitrates as low as 16 kbit/s per channel.

In order to limit the perceptible coding artifacts of common audio coding systems to a subjectively acceptable level, the entropy of the source has to be limited and the coding gain has to be optimized.

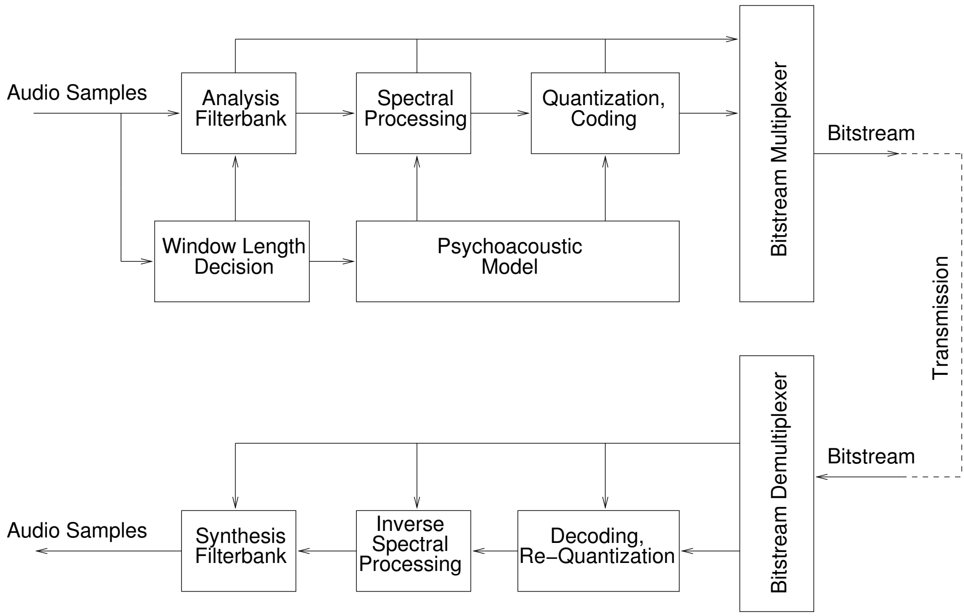

Fig. 1: Simplified overview of an MPEG-2/4 AAC Codec, as specified in [3][1].

This is generally achieved by reducing the coded audio bandwidth and the sampling frequency. To overcome this limitation, the SBR decoder reconstructs higher frequency components with the help of the low-frequency base band and a very compact parametric description of the high band [9][10]. The low-frequency base band of the signal is coded by a conventional core coder. In addition to that, the high band is dealt with by a Quadrature Mirror Filterbank (QMF) with 64 channels from which the SBR data is derived. Figure 2 illustrates the coding process. A detailed description can be found in [9].

Naturally, a combination of the abilities of HE-AAC and AAC-LD appears quite appealing in order to achieve a low bitrate and low delay coding system with high audio quality. In the following a coder of this nature is presented.

3. ENHANCED LOW DELAY AAC

This section describes the combination of the SBR tool with AAC-LD resulting in a low bitrate and low delay audio coding system. Several modifications are necessary to reach this goal.

3.1. SBR Framing

SBR, standardized in MPEG-4 as part of HE-AAC (see Section 2.3), has been defined for frame lengths of 1024 and 960 samples. However, AAC-LD uses 512 or 480 samples per frame. In a dual rate configuration, where the core codec runs at only half the input sampling frequency, a combination of AAC-LD with the SBR tool would hence cause the necessity of combining two AAC-LD frames with one SBR frame and thus introduce unnecessary delay. Therefore the frame length of the SBR module has been adapted to that of AAC-LD in the AAC-ELD coder.

3.2. SBR HF Reconstruction

The regular SBR decoder introduces an additional delay of six QMF slots due to the possibility of a variable time grid. This allows a non-synchronous distribution of the SBR parameter sets (envelopes) with respect to the core coder's framing grid. Removing the additional delay at the SBR decoder implies a locked time grid with synchronized envelope starts and endings. The regular SBR version is able to handle a transient at the end of a frame with one envelope only (see Figure 3).

With adapted, fixed framing grid, the coding of transients that occur at the end of an SBR frame has to be done in two subsequent short envelopes as shown

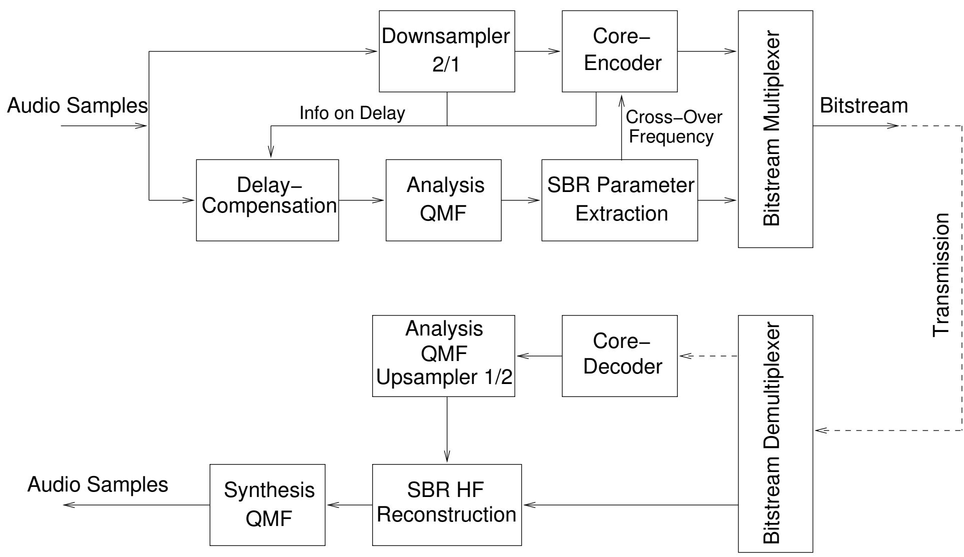

Fig. 2: Overview of SBR Codec in combination with a core coder, as specified in [8].

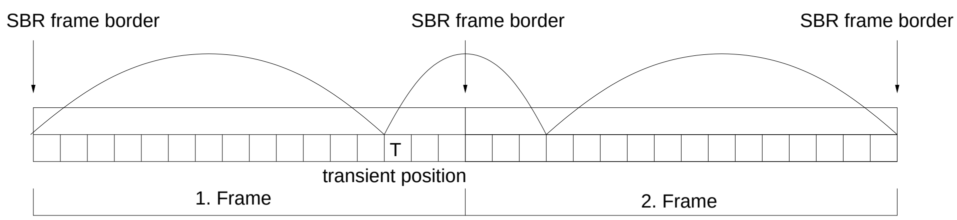

Fig. 3: Transient handling for classic SBR

in Figure 4. In this case, a higher number of side information sets is used to describe the transient correctly.

In the following, the SBR tool including these two modifications will be referred to as “SBR-LD”.

3.3. Low-delay Filterbank

Substantial delay reduction is achieved by utilizing a different window function with multiple overlap instead of the MDCT / IMDCT, thus obtaining a low-delay filterbank with perfect reconstruction.

3.3.1. History and Classification

The purpose of low-delay filterbanks is to reduce their reconstruction delay without reducing the filter length, but still maintain the perfect reconstruction property. This cannot be done with traditional filterbanks, like the TDAC filterbanks [11] or the MDCT. They are so-called para-unitary or orthogonal filterbanks, and that property results in symmetric windows and in the system delay being identical to the window length minus one.

Some of the first low-delay filterbanks were described in [12, 13] in the context of a generalized system delay, i.e. the system delay was no longer connected to the filter length. [12] described a direct design method via a numerical optimization. This approach did not guarantee perfect reconstruction and offered no simple way to obtain a fast implementation. [13] describes an optimization method for cosine modulated filter banks. While this leads to a considerably more efficient implementation, perfect reconstruction still was not a feature.

The design method used here was first described in [14, 15], and later in [16, 17] combining the de-

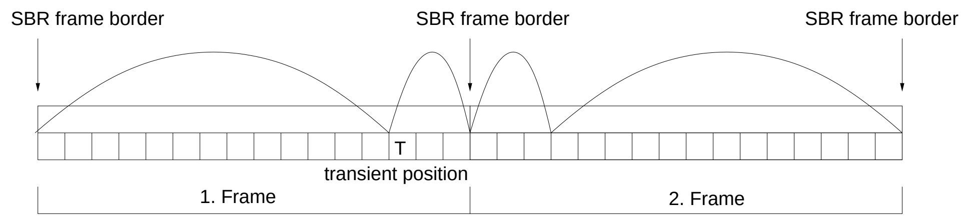

Fig. 4: Transient handling for low-delay SBR

sired properties. The resulting filterbanks have the same cosine modulation function as the traditional MDCT, but can have longer window functions which can be non-symmetric, with a generalized or low reconstruction delay.

3.3.2. Filterbank Windows

The new low-delay window for a frame size $M = 480$ samples reduces the MDCT delay from 960 samples $(2 \cdot M)$ to 720 $(2 \cdot M - \frac{M}{2})$ samples. Figure 5 shows the new window function and, for comparison, the traditional sine window. Note that the analysis window is simply a time-reversed replica of the synthesis window.

In the analysis window, the part that accesses future input values (and thus would cause delay) is reduced by 120 samples. Correspondingly, in the synthesis window the overlap with past output samples, which is needed in order to complete the overlap-add operation, is reduced by another 120 samples, resulting in an overall delay reduction of 240 samples. The extended overlap does not result in any additional delay, as it only involves adding values from the past.

The low-delay window provides a frequency response similar to that of the sine window, as can be seen in Figure 6. Figure 7 provides a comparison of the frequency responses of the low-delay window and the low-overlap window. It becomes obvious that the low-delay window has a much better frequency response compared to the low-overlap window.

The low-overlap window was introduced in [6] in order to eliminate pre-echo artifacts for transients. The lower overlap avoids a spreading of the quantization noise before the signal's attack. The new low-delay window has the same property, but offers a better frequency response. Therefore the low-delay window replaces both traditional AAC-LD windows, i.e. the sine and the low-overlap window and a dynamic window shape adaption is not necessary anymore.

3.3.3. Mathematical Description

We will use the following notation and symbols:

n	=	sample index
k	=	frequency index
i	=	window/block index
N	=	two times block length (equals $2 \cdot M$) length of traditional sine window
X	=	frequency samples
x	=	time samples
z	=	windowed samples
w_s	=	synthesis window coefficients

The analysis filterbank can be described similarly to the notation for the MDCT in MPEG [1]:

- windowing
  $z_i(n) = w_s(N-1-n) \cdot x(n),$
  for $n = -N, ..., N-1$

  The extension of the lower boundary down to $-N$, into past samples, accommodates the longer filter.

- analysis modulation

$$X(k) = -2 \sum_{n=-N}^{N-1} z_i(n) \cos[\frac{2\pi}{N}(n + \frac{1}{2} - \frac{N}{4})(k + \frac{1}{2})], \quad \text{for } k = 0, ..., \frac{N}{2} - 1 \tag{1}$$

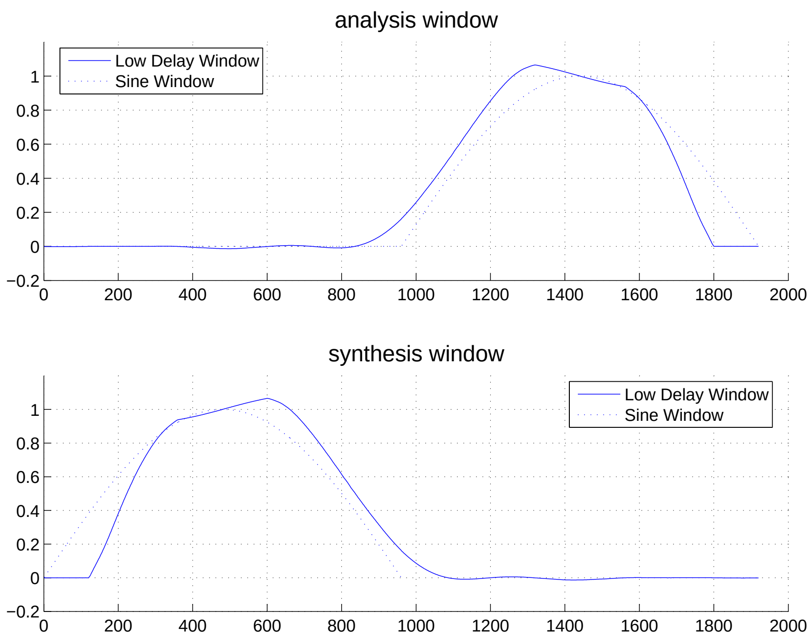

Fig. 5: Impulse response of low-delay windows

The synthesis can be described as follows:

- synthesis modulation

$$x(n) = -\frac{2}{N}\sum_{k=0}^{\frac{N}{2}-1} X(k)\cos[\frac{2\pi}{N}(n+\frac{1}{2}-\frac{N}{4})(k+\frac{1}{2})], \quad \text{for } n = 0, ..., 2N-1 \tag{2}$$

- windowing
  $z_i(n) = w_s(n) \cdot x(n)$, for $n = 0, ..., 2N-1$
- overlap-add
  $out_i(n) = z_i(n) + z_{i-1}(n+\frac{N}{2}) + z_{i-2}(n+N) + z_{i-3}(n+\frac{3N}{2})$, for $n = 0, ..., \frac{N}{2}$

4. DELAY ANALYSIS

This section provides an overview of the delay sources in the coding system and a comparison to other standardized codecs.

4.1. Delay Sources

Several sources of delay exist in the AAC core coder as well as in the SBR module. For a thorough discussion see [6]. For all further delay considerations a packet based transmission is assumed which causes no additional delay for the usage of a bit reservoir, as discussed in [18].

4.1.1. AAC Core

For AAC-LD the algorithmic delay can be described as $t_{LD} = 2 \cdot M$ samples. The low-delay filterbank reduces the number of samples by $M/2$ (see 3.3). In case of the usage of an AAC core in combination with SBR the delay is doubled due to the sampling rate conversion of a dual rate system.

4.1.2. SBR

Refering to Figure 2 two delay sources can be identified in the SBR decoder.

- QMF: The filterbank's reconstruction delay t_{SBR-fb} consists of 640 samples. Since the framing delay of $64-1$ samples is already introduced by the core coder, it can be subtracted to

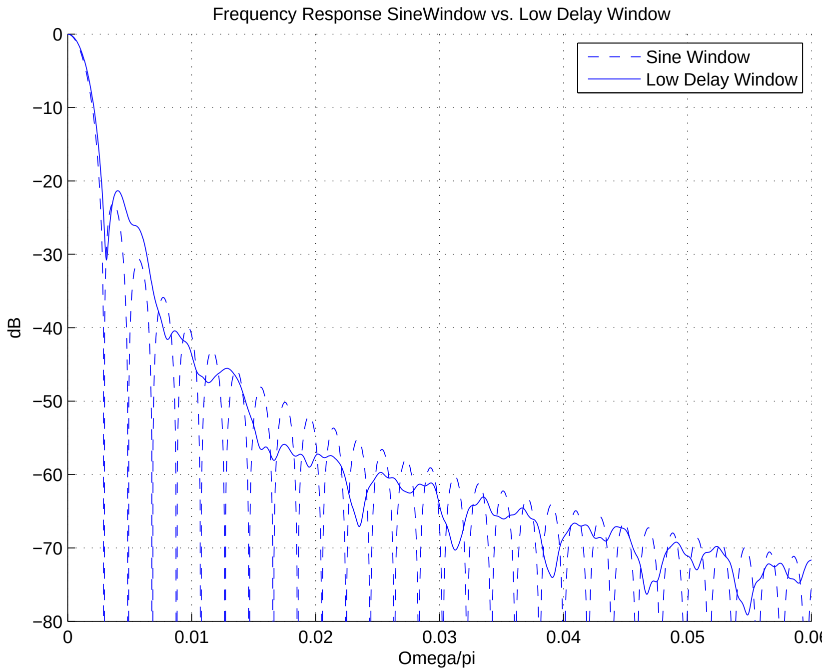

Fig. 6: Frequency response of sine window vs. low-delay window

obtain the delay value: $t_{SBR-fb} = 640 - 63 = 577$ samples.

- SBR HF reconstruction: As already mentioned in Section 3.2 the unmodified version of SBR causes an additional delay of six QMF slots due to the variable time grid: $t_{SBR-OL} = 6 \cdot 64 = 384$ samples.

4.1.3. Overall Delay

Table 1 provides an overview of the delay with the different modification stages assuming a frame length of 480 samples and a sampling rate of 48 kHz.

It can be seen that the combination of the described delay reduction methods indeed results in a delay saving of 18ms, i.e an overall algorithmic delay of 42ms rather than 60ms for the straight forward combination of AAC-LD and SBR.

4.1.4. Comparison with other Codecs

The algorithmic delay of the new AAC-ELD codec is compared to the most relevant codecs from MPEG as well as ITU-T communication codecs. Table 2 shows the typical delay values of several MPEG codecs.

Codec	Delay Source	delay [samples]	delay [ms]
AAC-LD + SBR	MDCT/IMDCT/ dualrate	$960 \cdot 2$	40
	QMF	577	12
	SBR-Overlap	384	8
		2881	60
AAC-ELD	low-delay filterbank/ dualrate	$720 \cdot 2$	30
	QMF	577	12
	SBR-Overlap	0	-
		2017	42

Table 1: Delay values for modifications on AAC-ELD

Table 3 shows the algorithmic delay of serveral ITU-T codecs.

The data shows that the presented codec's delay value lies well in the range of classic communication codecs'.

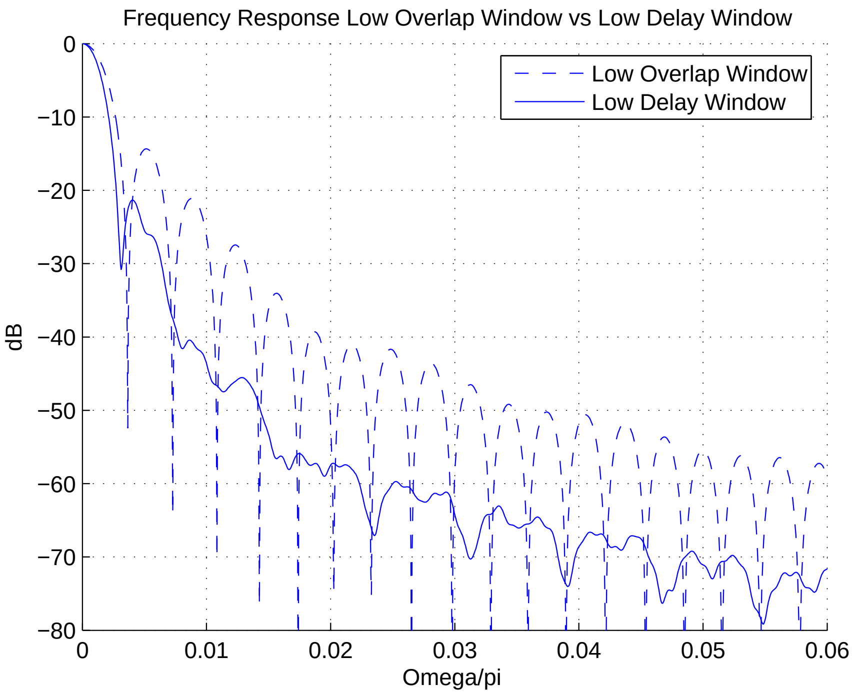

Fig. 7: Frequency response of low-overlap window vs. low-delay window

Codec	Sampling Rate	delay [samples]	delay [ms]
AAC-LC	32-48	2624	82-55
HE-AAC	48	6208	129
AAC-LD	24-48	960	40-20
AAC-ELD	48	2017	42

Table 2: Algorithmic delay values for MPEG codecs

Codec	Delay[ms]
AAC-ELD	42
G.729	15
G.722.1-C	40
G.722.2 (AMR-WB)	25
G.723.1	37.5

Table 3: Algorithmic delay of AAC-ELD compared to ITU-T codecs

5. PERFORMANCE EVALUATION

This section provides the results of listening tests conducted in order to assess the performance of AAC-ELD. Table 4 lists the items used in the tests.

5.1. Comparison of SBR and SBR-LD

A MUSHRA [19] listening test was carried out in order to assess the audio quality of the Low Delay SBR module including half frame length (see Section 3.1) and removed SBR overlap (see Section 3.2). The modified system was compared to a reference system as used in HE-AAC (see Section 2.3). In order to exclude the influence of (possibly different) core coders, the comparison was carried out by using a plain PCM signal as input. This input signal was band-limited to the SBR cross-over frequency after the SBR parameter extraction (encoding). The results of the comparison test can be seen in Figure 8. The test was taken by 10 experienced listeners in a high-quality listening environment.

As can be seen from the results, the low-delay version of the SBR module performs at least as good as the reference system. For some items a statistically significant improvement can be observed which can be explained by a more precise parameter extraction

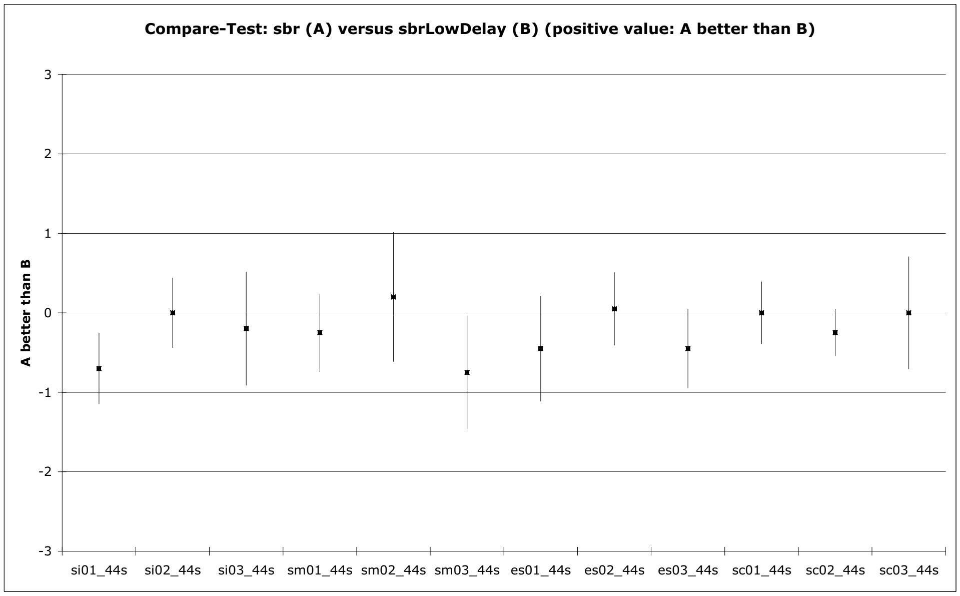

Fig. 8: Result of a comparison listening test between (A) unmodified SBR and (B) low delay version of the SBR module

Test signal	Content
es01	Suzanne Vega
es02	German male speaker
es03	English female speaker
sc01	Trumpet solo & orchestra
sc02	Symphonic orchestra
sc03	Contemporary pop music
si01	Harpsichord
si02	Castanets
si03	Pitch pipe
sm01	Bagpipes
sm02	Glockenspiel
sm03	Plucked strings

Table 4: Set of MPEG test items

due to a higher time resolution.

5.2. MPEG-4 ER AAC LD vs. Enhanced AAC LD

For assessing the performance of the AAC-ELD coder, a MUSHRA [19] listening test was carried out. For all coders in the test, an AAC-LD coder was used as the underlying coder (configured conforming to the MPEG-4 LD AAC profile Level 1, see [20]). The results of the test (taken by 9 experienced listeners) are shown in Figure 9. The list of codecs included

- usual reference and anchor conditions (codecs #1, #2, #3)
- the standard AAC-LD coder at bitrates of 32 kbit/s and 48 kbit/s (codecs #4, #5)
- the AAC-ELD codec at a bitrate of 32 kbit/s (codec #6)
- two additional check points (AAC-LD + low-delay filterbank, codec #7; and AAC-LD + SBR-LD, codec #8)

The following conclusions can be drawn from these results:

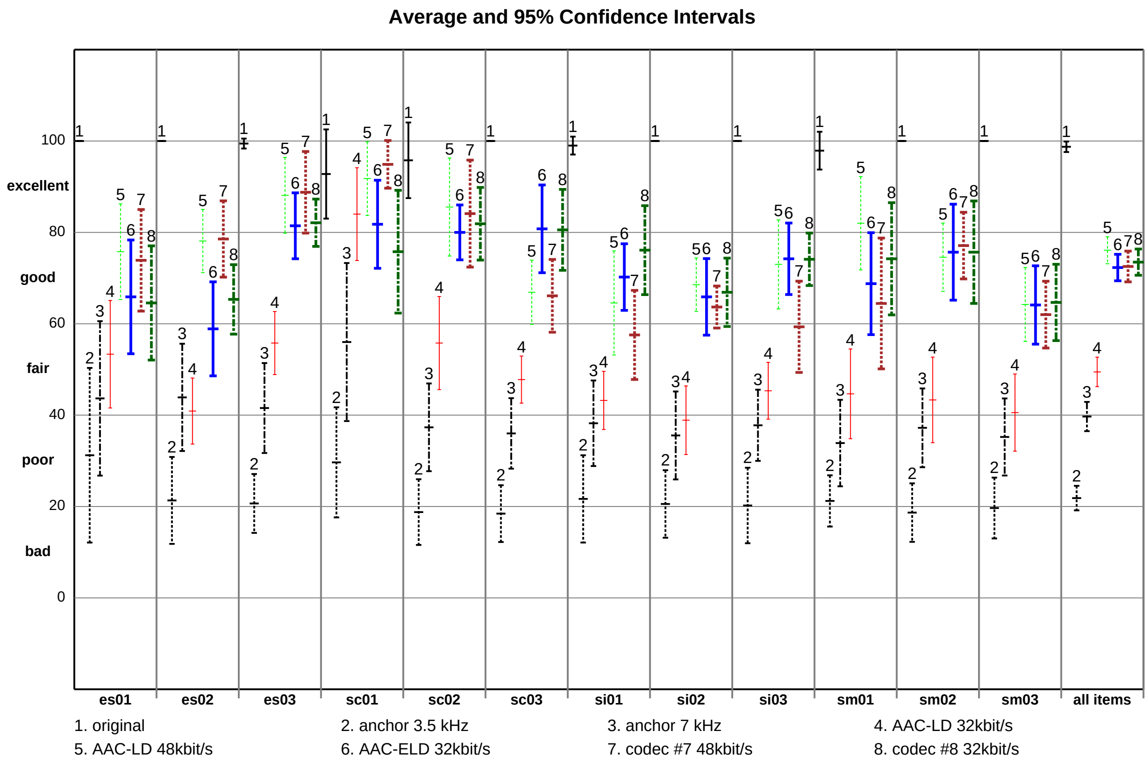

Fig. 9: Result of MUSHRA test

- The AAC-ELD coder at 32 kbit/s performs significantly better than the original AAC-LD coder at 32 kbit/s (codec #6 vs. #4).

- The AAC-ELD coder at 32 kbit/s performs statistically indistinguishable from the original AAC-LD coder at 48 kbit/s (codec #6 vs. #5).

- The check point coder combining AAC-LD and the low-delay filterbank performs statistically indistinguishable from the original AAC-LD coder, both running at 48 kbit/s (codec #7 vs. #5). This confirms the appropriateness of the low-delay filterbank.

- The check point coder combining AAC-LD and the SBR-LD at 32 kbit/s performs statistically indistinguishable from the original AAC-LD coder running at 48 kbit/s (codec #8 vs. #5). This confirms the appropriateness of the SBR-LD and dual rate approach.

6. IMPLEMENTATION ASPECTS

This section examines a number of implementation aspects and gives complexity estimates for the proposed AAC-ELD oder.

6.1. Low-delay filterbank

In the following, the computational complexity is derived relative to the complexity of the state-of-the-art AAC-LD codec. The main novelty is the low-delay filterbank which is examined with respect to its complexity in the following.

6.1.1. Computational complexity

The computational complexity of the IMDCT for AAC-LD with a frame size of $M = 512$ is derived in Table 5.

The low-delay filterbank can be implemented as efficiently as a regular MDCT (see also [17]) using the general structure as illustrated in Figure 10. The inverse DCT-IV and the inverse windowing/overlap-add are performed in the same way as for the traditional windows. $M/4$ window coefficients are zero,

M = 512		Instructions	
First modulation	$2 \cdot M$		1024
Complex FFT size	$M_2 = \frac{M}{2}$	256	
Number of Bfys	$\frac{M_2}{2}$	128	
Operations per Bfy	6	6	
Number of stages	$log2(M_2)$	8	
Total = $6 \cdot log2(M_2) \cdot \frac{M_2}{2}$		6144	6144
Second modulation	$2 \cdot M$		1024
Window and overlap-add	$2 \cdot M$		1024
Total			9216

Table 5: Arithmetic complexity of IMDCT + windowing (sine window)

M = 512		Instructions	
First modulation	$2 \cdot M$		1024
Complex FFT size	$M_2 = \frac{M}{2}$	256	
Number of Bfys	$\frac{M_2}{2}$	128	
Operations per Bfy	6	6	
Number of stages	$log2(M_2)$	8	
Total = $6 \cdot log2(M_2) \cdot \frac{M_2}{2}$		6144	6144
Second modulation	$2 \cdot M$		1024
Window and overlap-add	$2.75 \cdot M$		1408
Total			9600

Table 6: Computational complexity of IMDCT + windowing (low-delay window)

and thus do not involve any operation. For the by $2 \cdot M$ extended overlap into the past, only M additional multiply-add operations are required. In [17] these additional operations are referred to as "zero-delay matrices". From publications in the area of integer filterbanks these operations are also known as "lifting steps" [21]. The resulting overall number of operations is summarized in Table 6.

In summary, the complexity of the core coder including the low-delay filterbank is essentially comparable to that of AAC-LD using the regular MDCT/IMDCT filterbank.

6.1.2. Memory Requirements

In Tables 7 and 8 the RAM and ROM requirements for AAC-LD, AAC-LC and AAC-ELD including the low-delay filterbank (see 3.3) are compared. It can be seen that the memory increase for the low-delay filterbank is only moderate. The overall memory

Codec	frame length	working buffer	state buffer	sum [words]
AAC-LD	512	512	256	768
AAC-ELD	512	512	256+512	1280
AAC-LC	1024	1024	512	1536

Table 7: Comparison of RAM requirements

Codec	frame length	window coefficients	sum [words]
AAC-LD	512	$2 \cdot 512$	1024
AAC-ELD	512	$3 \cdot 512 - \frac{512}{4}$	1408
AAC-LC	1024	$2 \cdot 1024 + 2 \cdot 128$	2304

Table 8: Comparison of ROM requirements

requirement is still much lower compared to AAC-LC.

6.2. Real-time performance

Low-delay audio communication codecs are often run on fixed point processors because of their low power consumption, relatively low costs and widespread adoption in video conferencing systems. Table 9 shows performance measurements of AAC-ELD running on some popular fixed point processors. Since the fixed point implementation used is extremely portable in its nature, it runs on almost any fixed point processor that can perform 16 bit operand multiplications with a 32 bit result. For example ADI Blackfin, ARM9, TMS320C64xx, MIPS, PowerPC, etc. (see [22] Section 3 for more details). The implementation used did not involve any special optimization, because the intention is to show relative workload numbers. Absolute values would depend too much on the particular processor and implementation details. Some measurements were done on a selection of fixed point processors of this kind. An ADI Blackfin STAMP BF533 system running uCLinux at 500 MHz, an ARM XScale IXP420 system running at 266 MHz, an ARM RealView AXD Simulator ARM9E-S build and a Texas Instruments TMS320C6416 where used as test targets, and the coder processes a mono audio signal at a sampling rate of 48 kHz. The bitrate for AAC-ELD and HE-AAC is 32 kbit/s.

The result shows that the workload requirement of an AAC-ELD codec is very close to that of an HE-AAC codec. The memory requirement for a com-

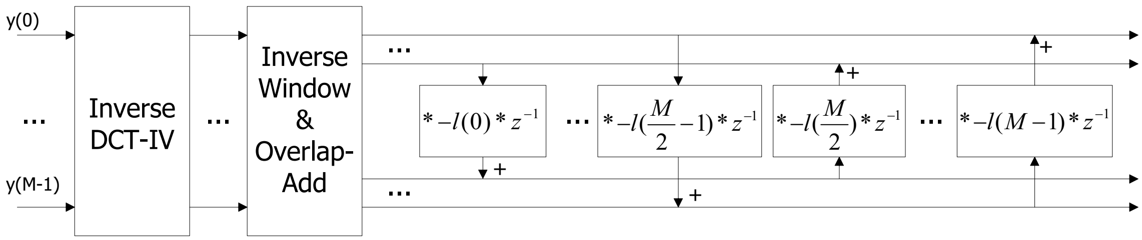

Fig. 10: Illustration of computationally efficient implementation of low-delay filter bank

Processor	AAC-ELD	HE-AAC
ADI Blackfin	98 / 117	100 / 100
ARM XScale	110 / 96	100 / 100
ARM RealView	92 / 115	100 / 100
TMS320C6416	106 / 93	100 / 100

Table 9: Realtime performance comparison (encoder/decoder workload in percent)

plete AAC-ELD codec filterbank proved to be as expected, just a little higher than a regular MDCT because of the longer window (see Table 8) and thus longer overlap-add buffers for the filterbank (see Table 7).

7. APPLICATIONS

Promising application scenarios for the new AAC-ELD codec are high fidelity video-/teleconferencing and Voice over IP applications of the next generation. This includes the transmission of arbitrary audio signals, e.g. speech or music or in the context of a multimedia presentation, at highest quality levels and competitive bitrates. The low algorithmic delay of AAC-ELD makes this codec an excellent choice for all kinds of communication applications.

8. CONCLUSIONS

In this paper the Enhanced MPEG-4 Low Delay AAC codec has been introduced. AAC-ELD is a combination of the AAC-LD codec and the SBR tool. With the incorporation of a low-delay filterbank and a modified SBR tool, the overall algorithmic delay is reduced so far that AAC-ELD develops high potential for communication applications. At the same time AAC-ELD achieves a much higher coding efficiency than classic AAC-LD. The field of applications for AAC-ELD include video-/teleconferencing and Voice over IP. The ongoing ISO/IEC MPEG standardization process of AAC-ELD is expected to be finalized by the end of 2007.

9. REFERENCES

[1] ISO/IEC 14496-3:2001, "Coding of Audio-Visual Objects, Part 3 Audio", 2001

[2] ISO/IEC 11172-3, "Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s, Part 3: Audio", 1992

[3] ISO/IEC 13818-3, "Information Technology - Generic Coding of Moving Pictures and Associated Audio, Part 3: Audio", 1994

[4] ISO/IEC 13818-7, "Information Technology - Generic Coding of Moving Pictures and Associated Audio, Part 7: Advanced Audio Coding", 1997

[5] M. Bosi, K. Brandenburg, S. Quackenbush, L. Fielder, K. Akagiri, H. Fuchs, M. Dietz, J. Herre, G. Davidson, Y. Oikawa, "ISO/IEC MPEG-2 advanced audio coding", *J. Audio Eng. Soc.*, Vol. 45, No. 10, pp. 789-813, Oct. 1997

[6] E. Allamanche, R. Geiger, J. Herre, T. Sporer, "MPEG-4 Low Delay Audio Coding Based on the AAC Codec", *106th AES Convention*, Munich, Germany, May 1999

[7] J. Hilpert, M. Gayer, M. Lutzky, T. Hirt, S. Geyersberger, J. Hoepfl, R. Weidner, "Real-Time Implementation of the MPEG-4 Low Delay Advanced Audio Coding Algorithm (AAC-LD) on Motorola DSP56300", *108th AES Convention*, Paris, France, Feb. 2000

[8] ISO/IEC 14496-3:2001/Amd.1:2003, "Coding of Audio-Visual Objects - Part 3: Audio, Amendment 1: Bandwidth extension", 2003

[9] M. Dietz, L. Liljeryd, K. Kjörling, O. Kunz, "Spectral Band Replication, a Novel Approach in Audio Coding", *112th AES Convention*, Munich, Germany, Apr. 2002

[10] P. Ekstrand, "Bandwidth Extension of Audio Signals by Spectral Band Replication", *Proc. 1st IEEE Benelux Workshop on Model based Processing and Coding of Audio*, Leuven, Belgium, Nov. 2002

[11] J. Princen, A. Johnson, A. Bradley, "Subband/Transform Coding Using Filter Bank Designs Based on Time Domain Aliasing Cancellation", *ICASSP*, 1987, pp. 2161-2164

[12] K. Nayebi, T. Barnwell, M. Smith, "Design of Low Delay FIR Analysis-Synthesis Filter Bank Systems," Proc. Conf. on Info. Sci. and Sys., Mar. 1991.

[13] T. Q. Nguyen, "A Class of Generalized Cosine-Modulated Filter Bank", International Symposium on Circuits and Systems, San Diego, 1992, pp. 943-946

[14] G. Schuller and M. J. T. Smith, "A General Formulation for Modulated Perfect Reconstruction Filter Banks with Variable System Delay," NJIT 94 Symposium on Appl. of Subbands and Wavelets, Mar. 1994

[15] G. Schuller, M.J.T. Smith,"Efficient Low Delay Filter Banks", DSP Workshop,Yosemite, CA, Oct. 1994

[16] Gerald D. T. Schuller, Mark J. T. Smith: "New Framework for Modulated Perfect Reconstruction Filter Banks", IEEE Transactions on Signal Processing, Vol. 44, No. 8, Aug. 1996

[17] Gerald D. T. Schuller, Tanja Karp: "Modulated Filter Banks with Arbitrary System Delay: Efficient Implementations and the Time-Varying Case", IEEE Transactions on Signal Processing, Vol. 48, No. 3, Mar. 2000

[18] R. Geiger, M. Lutzky, M. Schmidt, M. Schnell, "Structural Analysis of Low Latency Audio Coding Schemes", *119th AES Convention*, New York, USA, Oct. 2005

[19] ITU-R Recommendation BS. 1543-1, Method for the subjective assessment of intermediate sound quality (MUSHRA), International Telecommunication Union, Geneva, Switzerland, 2001

[20] ISO/IEC 14496-3:2005/Amd.1:200X, "Coding of Audio-Visual Objects - Part 3: Audio, Amendment 1: Low Delay AAC profile", to be published

[21] I. Daubechies, W. Sweldens, "Factoring Wavelet Transforms into Lifting Steps", Tech. Rep., Bell Laboratories, Lucent Technologies, 1996

[22] M. Gayer, M. Lohwasser, M. Lutzky "Implementing MPEG Advanced Audio Coding and Layer-3 encoders on 32-bit and 16-bit fixed point processors", *115th AES Convention*, New York, USA, Oct. 2003