|  |  |
| --- | --- |
| **ITU – Telecommunications Standardization Sector**  STUDY GROUP 16 Question 6  **Video Coding Experts Group (VCEG)**  73th Meeting: 16-20 October 2023, Hannover, GER | Document VCEG-BT05 |

|  |  |  |  |
| --- | --- | --- | --- |
| Question: | Q.6/SG16 (VCEG) | | |
| Source: | **Christian Helmrich, Jonathan Pfaff, Heiko Schwarz, Detlev Marpe, Thomas Wiegand (Fraunhofer HHI)** | Tel:  Email: | [firstname.lastname @hhi.fraunhofer.de](mailto:thomas.wiegand@hhi.fraunhofer.de) |
| Title: | **Information on performance evaluation of audio codecs for 2-channel ECG data** | | |
| Purpose: | Information | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Introduction**

This contribution reports on an investigation into the identification of a reference codec for compressing biomedical waveform data with two channels, intended to serve as a benchmark for further research and development within the field of the general compression of biomedical waveform data. The identified audio transform codec, Extended HE-AAC, is publicly available in source form, on both encoder and decoder side. As explained in this document, the encoder can be operated in an MSE distortion measurement compatible configuration, bypassing most psychoacoustic optimizations. The compression performances of using Extended HE-ACC with and without psychoacoustic encoder optimizations are compared. Results on the compression performance of the lossless audio-codec TAK, operated in a mode where least significant bits (LSBs) are successively removed, are also reported.

1. **Usage of MPEG-D Extended HE-AAC**

As a lossy audio codec, MPEG-D Extended HE-AAC [1], by way of the open exhale [2] encoder and FFmpeg’s FDK AAC [3] decoder, was used since

* it was found to outperform similar codecs in medium-rate audio quality in at least two recent blind listening tests [4],
* optional code is provided in the exhale source which allows to disable psychoacoustic and parametric coding features.

This optional mode, which can be activated by checking out exhale commit 0b683be9 (Sep. 2023) or later, defining EE\_MORE\_MSE 1 in header file src/lib/exhaleEnc.h and SFB\_QUANT\_PERCEPT\_OPT 0 in header file src/lib/quantization.h before compilation and utilizing

exhale(.exe) # n 99 input.wav bitstream.m4a (2)

as the encoder command-line, allows exhale to be operated in a reasonably MSE optimized fashion, with as few Intra coded frames as possible.

Value # in (2) is a numerical constant defining exhale’s bit rate preset (through log-proportional variation of the quantization step size) and, in this analysis, was set to either 5 or 6.

Because # allows for only relatively coarse variation of the bit rate and at least four BPS vs. PRD points were desired for comparative evaluation purposes, additional operating points were generated by changing, in line 969 of lib/exhaleEnc.cpp,

s = \_\_max (1u + ((UINT32\_MAX / (eightShorts … (3)

as follows, thereby effectively halving the quantizer step size:

s = \_\_max (1u + ((INT32\_MAX / (eightShorts … (4)

# Usage of Beck’s TAK (Tom’s Audio Kompressor)

As lossless audio codec reference, Beck’s TAK (Tom’s Audio Kompressor) [5] was used since it supports arbitrary sample bit depths and can therefore be operated in a lossy compression setting by zeroing LSBs. This was exploited by successively zeroing out LSBs of the PCM audio data (by appropriate rounding of the initial PCM data), followed by re-encoding and decoding with TAK. In doing so, lossy compression at reduced bit rate could be achieved even with this lossless coding solution. Encoding preset *–pMax* was used which represents the strongest compression offered by TAK. Note that, while TAK decoding is supported by FFmpeg, the encoder remains, as of late 2023, proprietary and closed-source.

# Experimental results

The audio-codecs described above were run to code all 48 sequences from the MIT-BIH Arrhythmia Database, which were provided by DICOM experts. Each sequence of the latter dataset consist of channels with samples. The data can be obtained from the FTP-server described below.

As a distortion measure, following the practice suggested by DICOM experts, the percentage root mean square distortion (PRD) was used. Here, if is the *j*-th sample value (with ) of channel *i* (with ) and if is the corresponding reconstructed sample value after decoding a bitstream, the PRD is defined as

The MIT-BIH Arrhythmia Database dataset was converted into PCM Wave (.wav) audio format at 16 bit/sample.Since the dataset exhibits a maximum data word length of only 11bit, five zero-valued LSBs were appended per sample in order to reach the 16-bit sample depth. The audio sampling rate was set to 32 kHz for compatibility with both audio codecs, and any container related file header overhead, as reported by MediaInfo [6], was excluded during the bitrate calculations.

The bitrate was measured in terms of bits per sample (BPS), defined as

Figure 1 below depicts the BPS vs. PRD performance of the two audio codecs discussed in the previous section, with each data point representing the arithmetic average across all 48 sequences of the MIT dataset. For Extended HE-AAC, two curves are shown, one for the typical (i.e., psychoacoustically optimized) behavior of the exhale encoder applied on audio input, the other for the MSE optimized (i.e., PRD friendly) behavior applicable to non-audio input.

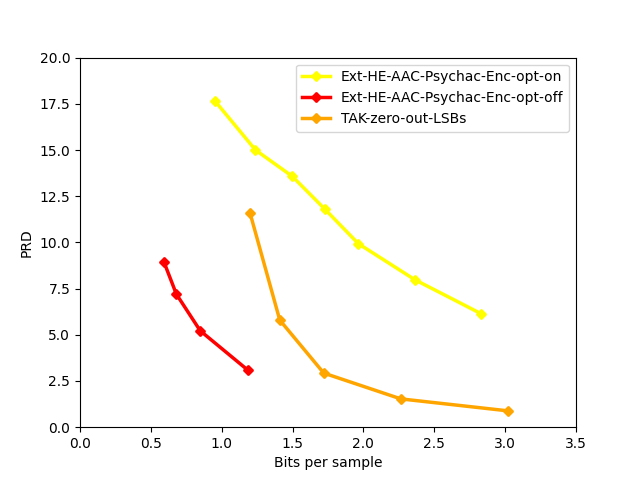


Figure 1: BPS versus PRD performance on the MIT-BIH Arrhythmia Database dataset of Extended HE-AAC, operated with and without psychoacoustic optimization, and for TAK.

# Availability of data and software

The test data as well as the software used to generate the above results for Extended HE-AAC can be downloaded from the following location:

Server: [ftp.hhi.fraunhofer.de](https://urldefense.com/v3/__http:/ftp.hhi.fraunhofer.de__;!!Ab1_Rw!ExOfc-x3p0e9RCmWEdTCbSatvCU5AIjYpC7ovhXAk9GmDH_epQgMsHc6UnSLIewoZqsap0lXQl4o7Z98T51cvaymJphwNcozfzg$)

Login: dicom

Password: yX5GUw.Zn

# References

1. ISO/IEC IS 23003-3, “Information technology – MPEG audio technologies – Part 3: Unified speech and audio coding,” Geneva, Jun. 2020.
2. C.R.Helmrich, project ecodis, “exhale: ecodis extended high-efficiency and low-complexity encoder,”version1.2, *Gitlab repository*, Sep.2023. <https://gitlab.com/ecodis/exhale>.
3. M. Storsjö, “A standalone library of the Fraunhofer FDK AAC code from Android,” *Git*, May 2022. Online: <https://github.com/mstorsjo/fdk-aac>, using FDK AAC v2, <https://www.iis.fraunhofer.de/en/ff/amm/impl.html>.
4. HydrogenAudio Forum, “Listening Tests,” Sep. 2020–2022. Online: <https://hydrogenaud.io/index.php/topic,121104.0.html> (user guruboolez) <https://hydrogenaud.io/index.php/topic,119861.0.html> (user Kamedo2).
5. T. Beck, “TAK,” version 2.3.3, July 2022. Online: [http://www.tbeck.de](http://www.tbeck.de/).
6. MediaArea.net SARL, “MediaInfo,” version 23.09, Sep. 2023. Online: https://mediaarea.net/en/MediaInfo

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_