Audio Engineering Society

Convention Express Paper 125

Presented at the 155th Convention
2023 October 25–27, New York, USA

Perceptually Motivated Bitrate Allocation for Object-Based Audio Using Opus Codec

Toni Hirvonen[1], Carlos Tejeda-Ocampo[2], Ema Souza-Blanes[1], and Sunil Bharitkar[1]

[1]*Samsung Research America*
[2]*Samsung Research Tijuana*

Correspondence should be addressed to Toni Hirvonen (t.hirvonen@samsung.com)

ABSTRACT

With the increasing popularity of immersive audio, using legacy tools for these new formats can be challenging. This paper presents an overview of how to utilize the Opus audio codec for object-based audio. We reviewed the performance of Opus using two different bit-allocation strategies: a vanilla method that uses the same bitrate for each object, and a joint allocation method that distributes the total bitrate among objects using their perceptual importance. The proposed joint allocation outperformed the vanilla method at half the total bitrate, and achieved an "Excellent" score during MUSHRA testing at 480 kbit/s for content with 20 objects.

1 Introduction

Object-based audio refers to a type of digital audio in which individual sound elements are represented as discrete, often mutually independent objects. These objects are accompanied by metadata that describe their relationships and properties. The metadata can vary over time and will be used by a renderer to generate appropriate signals for a target output layout. Alongside channel-based and scene-based audio, object-based audio is one of the types of audio signals considered for program reproduction in advanced sound systems [1].

Object audio offers several advantages over traditional channel-based audio and Ambisonics [2]. The main appeal for content creators and consumers is the promise that the same format can adapt to any output set-up. However, the format comes with a new set of challenges. For example, in the case of streaming audio, having a variable and often elevated number of audio elements to transmit (as opposed to a fixed set of audio channels) can be a problem from a datarate perspective. To address this problem, audio coding schemes have been proposed that target object-based audio, such as Spatial Audio Object Coding (SAOC) [3] and Joint Object Coding [4].

In addition to using novel tools, there is an incentive to handle object audio with existing and widely available legacy audio coding tools, such as the open-source, royalty-free Opus [5, 6, 7]. Opus is capable of low-latency and low-bitrate mono and joint stereo coding of speech and audio [8, 5, 6, 7, 9]. Opus encapsulation in the Ogg container supports mono-, stereo- and multi-channel surround cases (via joint stereo coding of multiple loudspeaker pairs) [10], as well as Ambisonics [11]. The performance of the Opus codec has been established for mono- and stereo audio [5, 6, 7, 9], multi-channel surround [12, 13, 14] and Ambisonics [15, 16, 17, 18, 11, 19, 20]. Current Opus coding and encapsulation does not specifically support object-based audio, and no research has been published on this topic.

The aim of this paper is to serve as an exploratory work on the viability of the current Opus codec for object-based audio, the effects of Opus compression on object-based rendering. In Sec. 2 we also propose a method for optimizing the Opus bitrate allocation, for example, on object content, and compare it against a vanilla baseline. The main contribution here is to present a computationally simple, yet psychoacoustically realistic stand-alone algorithm that can be adapted for preprocessing with any legacy codec, while remaining agnostic to the final rendering of the content. Sec. 3 presents the listening test methodology used, and Sec. 4 the analysis of the results.

2 Methods

2.1 Vanilla bit-allocation

The simplest solution for object coding with a traditional audio codec engine does not require any specific content analysis. Given a total bitrate b^{tot} and a number of objects N, the bitrate employed by the i^{th} object is given by

$$b_i = b_{tot}/N. \tag{1}$$

with $i \in [1...N]$. This method is used, e.g., in [21]

2.2 Motivation for joint bit-allocation

In immersive audio systems, objects have been predominantly utilized as spatial components [22, 23]. While formats like MPEG-H [22] allow for interactivity (e.g. changing the relative object playback gains, or listener position), most content creators are utilizing objects in a non-interactive context, expecting the overall audio mix to be faithful to the original version. In this scenario, auditory masking of less prominent objects can be exploited by assigning fewer bits to them.

The underlying psychoacoustic model of this work assumes that spatial release from masking is not very notable in a normal listening room [24], and that coding artifacts are mostly hidden by energetic, rather than informational masking [25]. This negates the need for pre-rendering, or complicated spatial analysis. Each object can be assumed to be masked by the sum signal of all audio. In other words, the goal of the proposed method is to remain agnostic to the listening setup, room acoustic properties, etc. Examples of tools that use this paradigm include methods that use frequency band parameters for least squares parametric object reconstruction from a downmix [3, 4].

A further assumption is that, typically, individual objects are not correlated to each other and there is no need to account for their covariances in coding to preserve the spatial impression. In practice, each object can be processed with a mono instance of an existing audio codec, as with the vanilla method of Eq. 1. As stereo coding is traditionally available in many existing codecs, it would be feasible to extend this to process e.g. correlated ambient objects. Such an extension, as well as the addition of simultaneous multichannel beds, is left for future work.

In order to perform optimal bit allocation and masking analysis between the object signals, there would be a need for frame-based time-frequency processing. Incorporating this into existing tools such as Opus would require developing and adopting a new version. In order to utilize existing versions of traditional audio coders as in Sec. 2.1, we propose a preprocessing analysis that approximates optimal joint bit allocation.

Finally, it should be noted that textbook auditory masking analysis alone is not the only factor for a competitive bit allocation in audio coding. For example, as the hearing sensitivity varies as a function of frequency, theoretical auditory masking models often utilize the equal-loundess contours [26] as a form of frequency weighting. However, this tends to de-emphasize lower frequencies that are important to modern music. Also, the decrease in temporal accuracy at higher frequencies enables very coarse parametric reconstruction of the fine structure [27, 6]. As a result, many codecs such as Opus utilize a combination of psychoacoustics and heuristics; in fact, the Opus baseline allocation is constant per frequency, only depending on the overall bitrate [6].

2.3 Proposed joint bit-allocation analysis

We assume a content consisting of two or more audio objects. For simplicity, the objects are further assumed to be monaural. As discussed in the previous Section, the goal is to optimize the bitrate assigned to each audio object. Before feeding it to the core codec (Opus), a preprocessing analysis is performed for a content segment. Although there is no limit on how short the duration can be, it is typically convenient to analyze a relatively long segment at a time. Typically, each input object segment is allocated a single bitrate parameter in the core codec interface that is applied for the whole duration of the object segment.

Each object signal is analyzed in perceptual frequency bands via Short-Time Fourier Transform (STFT), so that the frequency bins in each band are grouped together. The band frequency limits utilized in this paper are taken from the Opus spec, since we also use Opus as the core codec. However, STFT frame sizes do not need to match with the core codec MDCT windowing, but rather constant frame size of 2048 is used here. The purpose of the banding is to utilize a frequency-dependent weighting that mimics that used in Opus audio compression [6]. For each time-frequency tile of the the i^{th} time-domain object signal o_i, we can calculate the perceptually-weighted normalized energy, given the band perceptual weight:

$$e_i(t,k) = \frac{\alpha_k}{M}\sum_f |STFT(o_i)(t,f)|^2, \tag{2}$$

where k indicates the frequency band index, M the number of bins in the band, and α the predetermined perceptual weights. Operation $\sum_f$ sums over the bins within the frequency band k. Frequency weights α are here heuristic, and similar to the ones used in the Opus static bit allocation for the high rate (see Fig. 6 of [6]). Like Opus, we opt for static weighting instead of time-dependent masking analysis for simplicity, and due to the heuristic nature of bit allocation discussed in the previous section.

We propose using a measure of perceptual importance that takes into account both 1) the total frequency-weighted energy of the object signal E^{tot}, and 2) the average measure of how much the object signal is locally unmasked in time-frequency E^{mask}. The justification for the former is perhaps more intuitive; the objects that have more perceptual energy also are most critical to coding artifacts as they are masked least by the other objects. Also, since auditory masking is not an absolute threshold, we assign perceptual importance in a simple relative manner. For each object, we simply calculate and compare the sum of the weighted band energies over all time frames t and frequency bands k:

$$E_i^{tot} = \sum_t \sum_k e_i(t,k). \tag{3}$$

The second factor we utilize compares the frequency-weighted energy of the object to the frequency-weighted energy of the sum signal of all objects on the times the object is active, and calculates relative masking over time average. The goal is to obtain less artifacts if the object is listened to without any masking signals. For implementation, we apply Eq. 2 to all object signals individually, as well as to the total sum signal of all objects $s = \sum_i o_i$, resulting in the weighted energy $e_s(t,k)$ The latter represents the total masking signal that hides the local artifacts. The factor is given by:

$$E_i^{mask} = \sum_k mean_t(e_i(t',k)/e_s(t',k)). \tag{4}$$

We take the temporal average ($mean_t$) over all time frames of the ratio between the weighted signal energies of the individual object signal, and the sum masker signal. The reason for using averaging is that the factor of Eq. 4 only takes into account the time frames t' where each analyzed object is active, i.e. its energy is larger than some small silence threshold. The number of these active frames may vary between objects. The averaging process gives an indication how much the object tends to be locally unmasked throughout the entire content duration. This average is then summed over all frequency bands k to obtain the final measure.

Finally, we obtain the set of normalized factors in both cases by dividing the vector of individual object factors $E^{tot} = [E_1^{tot}...E_N^{tot}]$ and $E^{mask} = [E_1^{mask}...E_N^{mask}]$ by their respective sums i.e. L1-norms. We weight the factors Eq. 3 and Eq. 4 with relation of e.g. $w = 0.2E^{tot} + 0.8E^{mask}$ to have a final combined measure. We found that while only using the simpler total perceptual energy of Eq. 3 worked for some situations, it often results in critical failures when used alone for dynamically changing content, and that the factor of Eq. 4 should be weighted more for critical subjective testing.

2.4 Utilizing joint bit-allocation analysis

The pseudocode in Algorithm 1 shows a simple example of how to obtain the final assigned bitrates per object. This is based on normalized relative perceptual measures w, as well as three hyperparameters: 1) total available bitrate for all objects, i.e. whole content, 2) low limit bitrate per object, and 3) maximum bitrate per object. The latter two typically do not vary between the objects, but rather depend on the codec. Our example values were 6 and 64 kbit/s for Opus audio mode.

Once the rate hyperparameters are set, a simple iterative loop assigns the available bit reservoir at each step. In case there is overflow for some objects over the maximum rate, those bits are set as the new bit reservoir, and the process is repeated until the reservoir is depleted to a small value ε.

Algorithm 1 Get bitrates for N objects $[b_1 ... b_N]$

```
Require: weight per object $[w_1 ... w_N]$
Require: total bitrate $b^{tot}$
Require: low bitrate $b^{low}$
Require: high bitrate $b^{high}$
Ensure: $Nb^{low} \leq b^{tot} < Nb^{high}$
Ensure: $\sum^N w_i = 1$
  bitrate per object $b_i \leftarrow b^{low}$ for $i \leftarrow 1$ to $N$
  bit reservoir $r \leftarrow b^{tot} - \sum^N b_i$
  while $r > \varepsilon$ do
      for $i \leftarrow 1$ to $N$ do
          $b_i \leftarrow b_i + w_i * r$
          if $b_i > b^{high}$ then
              $b_i \leftarrow b^{high}$
              $w_i \leftarrow 0$
          end if
      end for
      $r \leftarrow b^{tot} - \sum^N b_i$
  end while
```

Although we experimented with purely object-based content for simplicity, many immersive formats can deal with channel-based audio (aka "bed") at the same time. Adding a simultaneous bed can be accounted for in principle; however, so far only by assigning separate bit budgets to the objects and beds; joint analysis of objects and beds if left for future work. Assuming the bed-portion is sent somehow, and give the remaining total rate for the objects, it is simple to apply Eq. 3 with no changes. Eq. 4 should however include the bed-part to the sum masker signal when calculating weights.

For real-time streaming, the question arises as to how the present method can best be utilized. Even though there are no theoretical limitations in analyzing short segments, total synchronization with the core codec framing and instantaneous bit allocation are not realistic due to the preprocessing nature of the method. However, even with real-time streaming, object-based content is typically available as a whole a priori; the content creator typically has mixed the audio and positional metadata of the objects as a whole for storage. Thus, the content can be analyzed in long segments before applying the legacy codec for real-time streaming.

3 Testing

3.1 Listening test methodology

A listening test was designed to assess the performance of Opus for objects, and specifically of the previously described joint bit allocation method (Sec. 2.3) by comparing an example of its application to the more straightforward vanilla method (Sec. 2.1). The S3A object-based audio drama dataset [28] contains 3 audio scenes, that were used as testing material. For each scene, two short segments were extracted upon informal listening assessments, resulting in 6 test tracks, as detailed in table 1.

To compare the two bit allocation methods, test stimuli were obtained by only including 20 objects from each track. These active objects were then coded as mono streams using Opus, and the bitrate dedicated to each mono object was determined using either the standard vanilla method or the joint allocation. Finally, the object- based scenes were rendered to a 7.1.4 speaker layout using the EBU ADM Renderer.

A MUSHRA[29] listening test was designed with the following bitrates:

- 480 kbit/s total, joint bit allocation method ("480k_alloc")
- 960 kbit/s total, vanilla method ("960k_vanilla")
- 480 kbit/s total, vanilla method ("480k_vanilla")

Scene	Content	Duration
Family1	Discussion in a house between family members, dishes noises	13.7s
Family2	Discussion in a house between family members, moving steps	9s
Forest1	Narrator, kids playing in the background, non-diegetic music	10.4s
Forest2	Narrator, water splashing, non-diegetic music	12s
Protest1	Discussion between coworkers in enclosed space, window being broken, distant sound from a protest	11.5s
Protest2	People chanting in a protest, helicopter flying above head	10s

Table 1: Listening test material

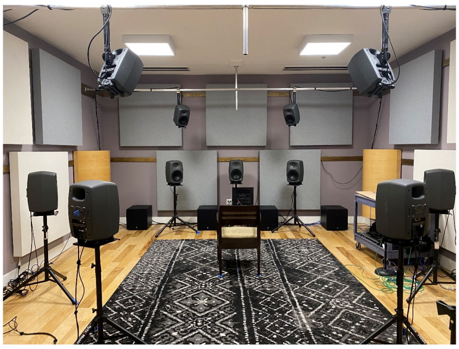

Fig. 1: Listening room

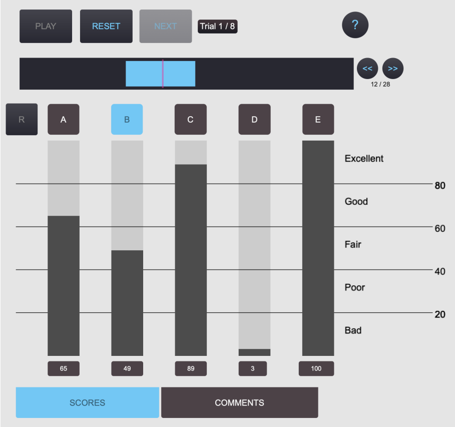

Fig. 2: Test user interface

3.2 Setup

Listening tests took place in a 7 m (L) x 5.33 m (W) x 3.05 m (H) listening room (Figure 1) equipped with 11 loudspeakers and 4 subwoofers supporting 5.1 and 7.1.4 channel-based playback. The positions of the speakers and the listeners were based on the ITU-R BS.2051-2 [30] standard. The loudspeakers were leveled at the listener's position and met the ITU-R BS.1116-3 [31] specification in terms of room response curve within the 50 Hz -16 kHz frequency range.

A tablet interface using Max/MSP (Fig. 2) provided the assessors with control over selection and playback of the test audios; as well as a rating module and the option to leave comments. A segment looping function enabled the assessors to focus on artifacts in restricted sections of the audio clips. The listening test software was implemented using a customized Max/MSP program to achieve a double-blind system, allowing for randomization of audio samples playback order and testing items mapping order on each trial.

4 Results

14 assessors completed 12 trials where the 3 test conditions were compared to the original uncompressed sample (reference), along with a hidden reference (H_Ref) and a low anchor (LA), and rated regarding Basic Audio Quality (BAQ). The panel of assessors was selected following the MUSHRA recommendation, as well as the ITU-R BS.2300-0 [32] method for the selection of assessors, and 3 listeners were excluded from the analysis.

Results show that both 480 kbit/s with bit allocation method and 960 kbit/s with vanilla method reached "Excellent" quality on the MUSHRA scale, while the 480 kbit/s with vanilla method condition only reached a "Good" quality. The 480 kbit/s with allocation method condition was rated slightly higher than 960 kbit/s with vanilla method. The same trend is repeated across the test samples (see Fig. 4).

The data were checked for homoscedasticity using Levene's test ($p = 3.25e^{-5}$), and the residuals for the linear model including the listening condition as factor were submitted to the Shapiro-Wilk test for normal distribution ($p = 1.65e^{-9}$). Since these two assumptions necessary for ANOVA were not met, Wilcox's robust alternative method for trimmed means [33] was employed to investigate the effects of the listening

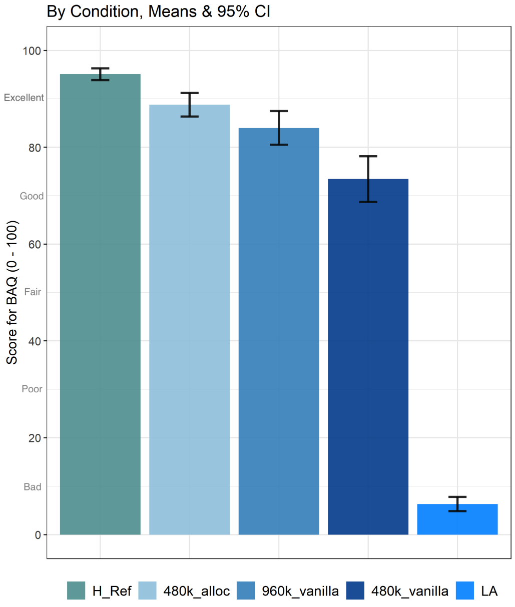

Fig. 3: Results for BAQ per bitrate

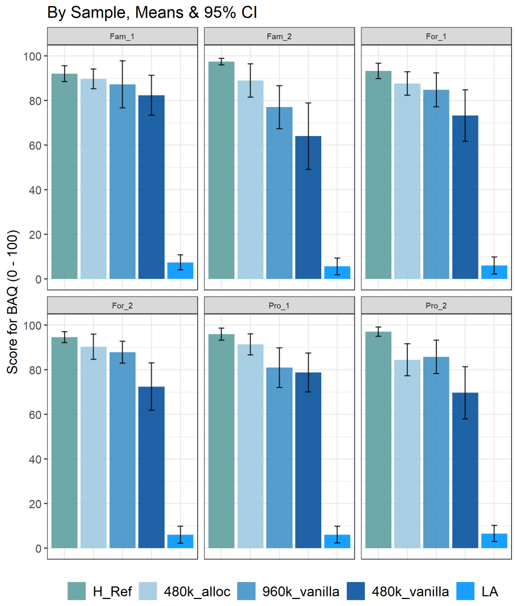

Fig. 4: Results for BAQ per bitrate per sample

	H. Ref	480k _alloc	960k _vanilla	480k _vanilla
480k _alloc	$p = 1.03e^{-4}$ $r = 0.361$			
960k _vanilla	$p = 1.80e^{-7}$ $r = 0.437$	$p = 4.44e^{-2}$ $r = 0.138$		
480k _vanilla	$p = 2.19e^{-10}$ $r = 0.627$	$p = 6.85e^{-5}$ $r = 0.420$	$p = 5.89e^{-7}$ $r = 0.285$	
LA	$p \sim e^{-13}$ $r = 0.866$	$p \sim e^{-13}$ $r = 0.864$	$p \sim e^{-13}$ $r = 0.864$	$p \sim e^{-13}$ $r = 0.864$

Table 2: Results of Wilcoxon pairwise comparisons (p-values p and effect sizes r); "small" effect sizes values in green color

condition and audio sample factors on the collected scores. The condition factor's effect was the only one found significant given a p value threshold of 0.05, with $p_{condition} = 0.001$, while $p_{sample} = 0.313$ and $p_{interaction} = 0.673$.

Since only the bitrate condition was found to have a significant effect on the BAQ scores distribution, paired-comparisons between each condition were performed using Wilcoxon's rank sum test. Effect sizes r and p-values p for each pair of conditions are reported in table 2.

None of the conditions was perceptually transparent to the hidden reference regarding BAQ. The samples that were compressed using the proposed bit allocation method (Sec. 2.3) to 480 kbit/s were rated significantly higher than both 480 kbit/s and 960 kbit/s with the vanilla approach. Smaller effect sizes are reported for relation of the *480 kbit/s with bit allocation - 960 kbit/s with vanilla method* and *480 kbit/s - 960 kbit/s with vanilla approach* pairs (See Table 2).

The statistical analysis shows that the bit-allocation method achieves a better basic audio quality at half the bitrate compared to the vanilla method.

5 Summary

To our knowledge, this is the first study on the feasibility of object-base audio using the Opus codec. Additionally, we introduced a method for computing the perceptual importance of audio objects present in a given content, in order to achieve efficient bitrate allocation for object-based audio with legacy audio codecs. We demonstrated the performance gains of this technique in comparison to a vanilla method (same bitrate for all audio objects) in a MUSHRA listening test. Our proposed method outperformed the quality of the vanilla alternative at half the total bitrate.

Thus, a significant bitrate reduction can be achieved with perceptually motivated allocation, and the overall rate can be kept realistic for streaming, at least when the number of objects remains reasonably low. While our allocation technique was tailored to the particularities of Opus compression, it could be adapted to work with other legacy codecs. Future work can extend these principles to stereo coding and multichannel beds. Legacy codecs could also benefit from other techniques, such as parametric coding and object-clustering, which could provide further improvements especially in cases where the number of objects is significantly higher.

Acknowledgements

Samsung Electronics and Samsung Research America supported this work. The authors would like to thank the entire staff of Samsung's US Audio Lab who helped with all aspects of this research, offered insightful suggestions, and contributed to this work.

References

[1] ITU-R BS.2051-3, "Advanced sound system for programme production," Standard, International Telecommunication Union, Geneva, CH, 2022.

[2] Bleidt, R., Borsum, A., Fuchs, H., and Weiss, S. M., "Object-based audio: Opportunities for improved listening experience and increased listener involvement," *SMPTE Motion Imaging Journal*, 124(5), pp. 1–13, 2015.

[3] Engdegård, J., Falch, C., Hellmuth, O., Herre, J., Hilpert, J., Hölzer, A., Koppens, J., Mundt, H., Oh, H.-O., Purnhagen, H., Resch, B., Terentiev, L., Valero, M. L., and Villemoes, L., "MPEG Spatial Audio Object Coding - The ISO/MPEG Standard for Efficient Coding of Interactive Audio Scenes," in *AES 129th Convention*, Audio Engineering Society, 2010.

[4] Purnhagen, H., Hirvonen, T., Villemoes, L., Samuelsson, J., and Klejsa, J., "Immersive Audio Delivery Using Joint Object Coding," in *AES 140th Convention*, Audio Engineering Society, 2016.

[5] Vos, K., Sørensen, K. V., Jensen, S. S., and Valin, J.-M., "Voice coding with Opus," in *AES 135th Convention*, Audio Engineering Society, 2013.

[6] Valin, J.-M., Maxwell, G., Terriberry, T. B., and Vos, K., "High-quality, low-delay music coding in the Opus codec," *arXiv preprint arXiv:1602.04845*, 2016.

[7] Hoene, C., Valin, J.-M., Vos, K., and Skoglund, J., "Summary of Opus listening test results," Technical report, IETF, 2011.

[8] Valin, J.-M., Vos, K., and Terriberry, T. B., "Definition of the Opus Audio Codec," RFC 6716, 2012, doi:10.17487/RFC6716.

[9] Hines, A., Gillen, E., Kelly, D., Skoglund, J., Kokaram, A., and Harte, N., "Perceived audio quality for streaming stereo music," in *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 1173–1176, 2014.

[10] Terriberry, T., Lee, R., and Giles, R., "Ogg Encapsulation for the Opus Audio Codec," RFC 7845, 2016, doi:10.17487/RFC7845.

[11] Skoglund, J. and Graczyk, M., "Ambisonics in an Ogg Opus Container," RFC 8486, 2018, doi: 10.17487/RFC8486.

[12] Trojahn, F., Meszaros, M., Maruschke, M., and Jokisch, O., "Surround sound processed by Opus codec: a perceptual quality assessment," *Studientexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung 2017*, pp. 300–307, 2017.

[13] Siegert, I., Jokisch, O., Lotz, A. F., Trojahn, F., Meszaros, M., and Maruschke, M., "Acoustic cues for the perceptual assessment of surround sound," in *International Conference on Speech and Computer*, pp. 65–75, Springer, 2017.

[14] Devantier, A., Tejeda-Ocampo, C., Zhongran Wang, C., Saba, W., and Bharitkar, S., "Bit Rate Requirements for an Audio Codec for Stereo,

Surround and Immersive Formats," in *AES 151st Convention*, Audio Engineering Society, 2021.

[15] Narbutt, M., O'Leary, S., Allen, A., Skoglund, J., and Hines, A., "Streaming VR for immersion: Quality aspects of compressed spatial audio," in *23rd International Conference on Virtual System & Multimedia (VSMM)*, pp. 1–6, IEEE, 2017.

[16] Narbutt, M., Skoglund, J., Allen, A., Chinen, M., Barry, D., and Hines, A., "Ambiqual: Towards a quality metric for headphone rendered compressed ambisonic spatial audio," *Applied Sciences*, 10(9), p. 3188, 2020.

[17] Rudzki, T., Gomez-Lanzaco, I., Hening, P., Skoglund, J., McKenzie, T., Stubbs, J., Murphy, D., and Kearney, G., "Perceptual evaluation of bitrate compressed ambisonic scenes in loudspeaker based reproduction," in *AES International Conference on Immersive and Interactive Audio*, Audio Engineering Society, 2019.

[18] Rudzki, T., Gomez-Lanzaco, I., Stubbs, J., Skoglund, J., Murphy, D. T., and Kearney, G., "Auditory Localization in Low-Bitrate Compressed Ambisonic Scenes," *Applied Sciences*, 9(13), p. 2618, 2019.

[19] Souza-Blanes, E., Tejeda-Ocampo, C., Wang, C., and Bharitkar, S., "Bitrate Requirements for Opus with First, Second and Third Order Ambisonics reproduced in 5.1 and 7.1. 4," in *AES 152th Convention*, Audio Engineering Society, 2022.

[20] Blanes, E. S., Tejeda-Ocampo, C., and Bharitkar, S., "A Closer Look on Bitrate Requirements for Opus with First Order Ambisonics and AllRAD rendering to 5.1 and 7.1. 4," in *AES 153rd Convention*, Audio Engineering Society, 2022.

[21] Pfanzagl-Cardone, E., "SONY 360 Reality Audio," in *The Art and Science of 3D Audio Recording*, pp. 267–277, Springer International Publishing, 2023.

[22] ISO/IEC 23008-3:2022, "Information technology - High efficiency coding and media delivery in heterogeneous environments - Part 3: 3D audio," Standard, International Organization for Standardization, Geneva, CH, 2022.

[23] ETSI TS 103 190 parts 1 2, "Digital Audio Compression (AC-4) Standard," Standard, European Telecommunications Standards Institute, Geneva, CH, 2018.

[24] Koenig, H., Allen, J. B., Berkley, D. A., and Curtis, T. H., "Determination of masking-level differences in a reverberant environment," *J. Acoust. Soc. Am.*, 61, p. 1374–1376, 1970.

[25] Kidd, G., Mason, C., Brughera, A., and Hartmann, W., "The Role of Reverberation in Release from Masking Due to Spatial Separation of Sources for Speech Identification," *Acta Acustica united with Acustica*, 91, pp. 526–536, 2005.

[26] Suzuki, Y. and Takeshima, H., "Equal-loudness-level contours for pure tones," *The Journal of the Acoustical Society of America*, 116(2), pp. 918–933, 2004.

[27] ISO/IEC 14496-3, "Information technology - Coding of audio-visual objects - Part 3: Audio - Amendment 1: Bandwidth extension," Standard, International Organization for Standardization, Geneva, CH, 2003.

[28] Woodcock, J., Pike, C., Coleman, P., Franck, A., and Hilton, A., "S3A object-based audio drama dataset," 2020.

[29] ITU-R BS.1534-2, "Method for the Subjective Assessment of Intermediate Quality Level of Audio Systems," Standard, International Telecommunication Union, Geneva, CH, 2014.

[30] ITU-R BS.2051-2, "Advanced Sound System for Programme Production," Standard, International Telecommunication Union, Geneva, CH, 2018.

[31] ITU-R BS.1116-3, "Methods for the Subjective Assessment of Small Impairments in Audio Systems," Standard, International Telecommunication Union, Geneva, CH, 2015.

[32] ITU-R BS.2300-0, "Methods for Assessor Screening," Standard, International Telecommunication Union, Geneva, CH, 2014.

[33] Wilcox, R., *Introduction to robust estimation and hypothesis testing. 3rd ed*, volume 93, 2012.