Audio Engineering Society

Convention Paper 10686

Presented at the AES 156th Convention
2024 June 15–17, Madrid, Spain

This paper was peer-reviewed as a complete manuscript for presentation at this convention. This paper is available in the AES E-Library (http://www.aes.org/e-lib), all rights reserved. Reproduction of this paper, or any portion thereof, is not permitted without direct permission from the Journal of the Audio Engineering Society.

A Hierarchical Approach to Lossless Coding of Multichannel and Immersive Audio

Toni Hirvonen[1], Mahmoud Namazi[1], and Carlos Tejeda-Ocampo[2]

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

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

ABSTRACT

Techniques for improving multichannel lossless coding are introduced. A method is proposed for hierarchical simultaneous coding of two or more different mixes of the same content. Unlike in prior methods limited to cascaded Toeplitz systems, the prediction parameters are jointly optimized via a general linear solver, and the prediction residual is Rice coded. Additionally, the use of a projection prior to residual coding is proposed as being of general benefit in lossless audio coding. Comparisons with various multichannel and immersive material are made against more traditional methods, including FLAC. The proposed methods show improved compression ratios for surround music alone, as well as for hierarchical coding. Future work would include further validation and improved modeling of a non-static downmix relation to upmix.

1 INTRODUCTION

As immersive audio gains popularity and content creators utilize the capabilities of these modern formats, the same content is increasingly available for different multichannel listening setups, such as 5.1 and 7.1.4 [1, 2]. Because the signals are often significantly correlated, there exists opportunities to leverage these correlations for simultaneous coding of multiple mixes of the same content, leading to enhanced compression.

While perceptual compression has produced important advancements in audio technology, lossy multichannel coding is challenging to optimize perceptually. Perceptual differences are difficult to judge due to their multidimensional nature [3]. While many codecs make good arguments that they achieve transparency after some bitrate, this cannot be fully guaranteed for all possible content due to the limitations of subjective testing. As transmission capacities have constantly improved, the need for extremely low bitrates is no longer as major a concern as in the past. Furthermore, prejudices against lossy coding and transmission have increased as both consumers and content creators become more educated. Similarly, although automatic rendering of object-based content [4], and blind upmixing of channel-based material [5] may achieve an approximation of the desired multichannel playback in different setups, there is no specific guarantee of preserving the artistic intent.

Lossless coding is a viable option to address concerns related to automatic rendering, upmixing, and lossy coding in situations where there is a need for exact control of the immersive audio reproduction process in all situations. The overall motivation of this paper is to examine the current status of multichannel lossless coding for this application.

Compared to traditional lossless coding, it is not as clear how to most effectively deal with the coding of multichannel audio by exploiting channel correlations. As we see in Sec. 2, a system using a very simple baseline model of coding channels separately is able to get very close to real codec performance. More sophisticated prediction models have been proposed in [6, 7, 8, 9]. However, in the case of multichannel audio redundancy, these models have not resulted in major benefits, but rather in small improvements [6], or the exact methods have not been publicly fully detailed [8]. (See Sec. 3.3.3 for further discussion.)

Our main novel contribution is to propose signal prediction models and optimization methods for an audio system, where several (two or more) different mixes are stored simultaneously for the same content. Having mixes in the same container can very effectively control the artistic intent for the content as well, regardless of compression. Alternatively, the downmix(es) can be assumed to be available a priori at the decoder. The method exploits correlations between the different content versions and reconstructs more elaborate presentations based on the lower-level formats. Related hierarchical lossless multichannel coding systems have been proposed before in [8], where efficient coding of stereo downmix, and an optional 5.1 mix was enabled from a 6-channel bitstream by sending the de-mixing matrix as additional side information. However, this system is limited to locally static and linear channel-combination downmixes, whereas our proposed models are capable of non-constrained hierarchical modeling.

We implemented controlled experiments showing the potential benefits of using more complex, globally optimized prediction models than in prior art [6, 7, 8]. We propose a method combining short-term prediction, Singular Value Decomposition (SVD) projection, and Rice coding. The different systems are compared with multichannel audio databases. In addition to benefiting hierarchical lossless coding, it performs considerably better than the baselines for normal lossless coding of 5.0 music, at an increased computational encoding cost. Details of the methods are presented in Sec. 2. The experiment results, and discussion about their implications follow in Sec. 3 and Sec. 4.

2 METHODS

2.1 Core lossless coding engine

Current real-life lossless codecs share many compression principles, techniques, as well as overall performance. FLAC (Free Lossless Audio Codec) [10] is an open source, widely-used codec implementation whose fundamental algorithms are based on the earlier Shorten [11] tool. In this paper, we apply the official FLAC implementation as a reference, and replicate its performance with a simple baseline implementation.

The general principle applied in FLAC and our method can be described with the following simplified signal model (time and channel indices omitted):

$$s = f(s') + e, \tag{1}$$

where the original signal s is represented by a *predictor function* $f()$ operating on a predictor source signal s', which is often related to s. The prediction residual is noted with e. In standard lossless coding, $f()$ is often a linear predictor (LPC) operated on short frames, giving the signal model:

$$s(t) = \sum_{k=1}^{p} \beta_k s(t-k) + e(t). \tag{2}$$

For each time sample of the frame, p (i.e. prediction order) past samples are used as linear combination to model it. The coefficients $\beta = [\beta_1 ... \beta_p]$ are typically solved for minimizing the frame residual MSE $||e||_2^2$. In real codecs, search procedures and frame signaling are often used to find the best p, as well as sometimes the type of predictor solution (e.g. LPC or a standard template [11]). In this paper, we omit this optimization step, and rather aim to isolate the effect of the predictor source s'.

The compression ratio achieved by lossless audio codecs is predominantly affected by the entropy coding of the prediction residual e. FLAC and Shorten simply assume that the residual distribution is geometric, with symmetrical focus around value zero. These will have some Golomb code [12] as an optimal prefix code. Rice coding [13] is a subset of such codes where the Golomb parameter is power of 2 for computational efficiency. We replicate the algorithm in the FLAC codebase [10] to calculate the codeword length in bits as a function of

the Rice parameter r. The optimal Rice parameter r can be estimated from the signal [11], but we used a brute-force search (e.g. $r \in [0...20]$) and selected the code that minimizes the sum amount of bits in the analysis frame of each channel.

Despite the simplicity of this baseline signal model, it already accounts for much of the performance of the current real-life lossless codecs (see Sec. 3.3). Some further tools, such as efficient handling of silent frames and signal runs is not considered here, but can certainly improve results for sparse material. Replacing Rice coding with arithmetic coding [14], or hybrid entropy coding [6] gains typically few percent in compression efficiency. We also experimented with an additional long-term predictor that attempts to find the best matching segment to the current frame from the full history of the signal [6], but did not include it in our models.

2.2 Multichannel modeling

With multichannel, or immersive audio, the question becomes whether correlations between channels can be exploited. In the case that s has more than one channel, one can use the model of Eq. 2 for each channel c separately; each sample $s_c(t)$ only is predicted from the past samples of that same channel. The number of model parameters is then pC, where C is the number of channels. In contrast, we also construct a vanilla baseline for a multichannel predictor in order to test the hypothesis that there are easily exploitable correlations between the channels:

$$s_c(t) = \sum_{c=1}^{C} \sum_{k=1}^{p} \beta_{c,k} s_c(t-k) + e_c(t). \quad (3)$$

In effect, the current sample of each channel is predicted using all other channels' samples looking p timesteps in the past. This increases the amount of model parameters to pC^2.

Another possibility for exploiting the correlations between channels is to utilize a *transform* with desirable properties. For example, a common technique is to use Principal Component Analysis (PCA) or SVD to find a linear projection of maximal energy compaction, and orthogonality of the transformed components [15, 16]. To our knowledge, this technique has not been well investigated for lossless multichannel audio coding, and it is only approximated with mid-side channel pairs and similar heuristics. The SVD projection method of [16] was utilized here. To avoid large values, we found that applying the projection to the prediction residual e and not to the original signal is preferable. It should be noted that while computationally complex at the encoder, the more crucial decoding cost of such transform is only increased by a single matrix multiplication.

2.3 Hierarchical reconstruction from downmix

The main contribution of this paper is to suggest a multichannel audio signal model, which when optimized, can be used for efficient prediction in the context of hierarchical reconstruction. Assume that the decoder has available many mixes of the same content, either from the same container, or otherwise. Decoding is traditionally done first on the lowest mix in the hierarchy (aka "downmix", typically the mix with the least amount of channels). It is then used to predict the next mix (aka "upmix") with the signal model. The whole process can be repeated by using the decoded upmix as the new downmix for the next iteration.

We test adding simple additional predictors that utilize the downmix to the previous single-, and multichannel models of 2 and 3:

$$s(t) = \sum_{k=1}^{p} \beta_k s(t-k) + \sum_{d=1}^{D} \gamma_d s_d(t) + e_c(t), \quad (4)$$

$$s_c(t) = \sum_{c=1}^{C} \sum_{k=1}^{p} \beta_{c,k} s_c(t-k) + \sum_{d=1}^{D} \gamma_d s_d(t) + e_c(t), \quad (5)$$

where s_d indicates channel d of the downmix, and γ_d the corresponding prediction parameter.

It can be seen that these models only utilize the most current sample of the downmix, in addition to predicting from the past of the upmix as in traditional models. We found this worked the best for our tests, as compared to more elaborate utilization of the downmix. Also, the addition of such downmix prediction only introduces $D*C$ more model parameters. Also important for the hierarchical models is the optimizer selection, as discussed in Sec. 2.4.

2.4 Model optimization

Traditionally, lossless coding predictors have been optimized with Levinson recursion [17, 18]. These methods achieve computational efficiency by assuming Toeplitz systems. The Toeplitz assumption however limits the type of predictors that are possible: all solved predictor parameters must originate from a time series of consecutive samples. Another alternative used by e.g. [6, 9] is to use several cascaded Toeplitz models whose parameters are not optimized globally. In matrix form, the minimization goal becomes:

$$argmin_{\alpha}||s - S'\alpha||_2^2 \tag{6}$$

where prediction source S' is a Toeplitz matrix with different lags of source signal s' as columns. In case of the standard single-channel model 2 of order p:

$$\alpha = [\beta_1, , \beta_p]. \tag{7}$$

In contrast, we use well-established solvers for linear systems that are not limited to be Toeplitz, namely the GELSD algorithm available in LAPACK [19]. Despite being slower computationally, it allows optimizing all model parameters globally when using the complicated models of Eq. 3 and Eq. 5, and include arbitrary columns to the source signal matrix S'. When the predictor is based on the model of Eq. 5, we have:

$$\alpha = [\beta_{1,1}, , \beta_{C,p}, \gamma_1, , \gamma_D], \tag{8}$$

for prediction order p, C upmix channels, and D downmix channels, respectively.

To enable comparison, GELSD is used for all prediction models. Computational efficiency refinements are largely left for future work. We however utilize Tikhonov regularization [20] in all solvers, except the single-channel baseline Eq. 2, by solving for smaller (covariance) matrices, and adding a diagonal component δI:

$$argmin_{\alpha}||S'^T s - (S'^T S' + \delta I)\alpha||_2^2. \tag{9}$$

As is typical, the 16-bit integer input signals are transformed into double precision float in the range $[-1, 1]$ for optimization computations, For simplicity, all solved model parameters are quantized as 16-bit floats prior to the residual calculation and Rice coding. Mirroring the datatype changes and rounding operations in the decoder ensures lossless reconstruction.

It should be noted that despite the similarity of some of our proposed signal models to neural networks, gradient descent - based algorithms did not work well here. For frame-based optimization, the amount of training data favors using a global optimization method like GELSD compared to any greedy optimization.

3 EXPERIMENTS

3.1 Datasets

3.1.1 5.1 Music Database

We obtained a custom dataset of 100 songs that had been mixed and mastered specifically to the 5.1 format in the past 10 years. The content included mainly pop/rock, and classical genres from various different performers. In our view, such an ad-hoc dataset represents a general, realistic situation, and the exact content or music style is not a determining factor for the overall performance.

The LFE channel is omitted in the dataset; we only use the 5.0-channels (L, R, C, Ls, Rs) of the mixes. Preliminary experiments indicated that including LFE in the prediction would not help, and thus sending it with a single-channel predictor like Eq. 2 would just add the same amount of bitrate for all the methods in the comparisons. Furthermore, LFE channel coding may benefit from advanced silence handling, which was not the focus of this paper.

We utilize the static ITU standard downmix [21] from 5.0 to 2.0 stereo in order to show the benefit of hierarchical reconstruction in the typical situation where the downmix is correlated to the upmix. Of course, this is an artificial situation; in real life this downmix could be obtained at the decoder without sending it, by applying the known linear operation of [21], as in [8]. However, we believe the results also indicate that there is a benefit when using an artistic downmix, especially if the processing in mixing consists of linear operations such as panning. This assumption may break down in rarer cases of strong nonlinearities or uncorrelated mixes. It should also be emphasized that we are not in this paper addressing object audio, but channel-based material. The former aims to be agnostic to the rendering setup by sending panning information per object, and thus can in principle account for the upmixing blindly.

3.1.2 7.1.4 Immersive Audio Database

Material for the 7.1.4 loudspeaker setup (standard 7.1 plus 4 height speakers) is not as readily available as for the 5.1 setup. Furthermore, for our tests obtaining all three different formats (stereo, 5.1, and 7.1.4) of the same content was of interest when testing immersive audio. To get the different representative renderings for the same content, we used object-based content from Netflix [22] and the S3A database [23]. The EBU ADM Renderer [24] was used to produce the format renderings. In total we selected 10 different clips of representative content from both databases.

To complement the object-based, more cinematic material, we added 9 commercial music pieces that were specifically mixed for 7.1.4. In order to get the different mixes of the same music content, we again utilized the static ITU standard downmix [21] from 7.1.4 to 5.1, and from 5.1 to stereo. Our initial observation was that compared to music, cinematic content typically has less total, and more sporadic activity in the channels. All other processing details, such as omitting the LFE, were as discussed in Sec. 3.1.1.

3.2 Systems tested

All material was utilized with 16 bit depth and 44.1 kHz sample rate. We compared FLAC against our proposed methods. The models used for prediction discussed in Sec. 2 had their parameters optimized with the Tikhonov regularized ($\delta = 1^{-4}$) GELSD solver for Eq. 9. For the basic single-channel model of Eq. 2, GESLD was used to optimize Eq. 6 in order to compare this baseline against FLAC with similar solver criterion. We used FLAC with the default parameters, as experimenting with other options resulted in little difference.

The prediction order for all models implemented (as well as the default FLAC LPC max order) was $p = 8$. Unlike in real-life coders, we used constant p for each frame of 4096 samples. The only hyperparameter signaled per frame was the Rice code parameter per channel. SVD projection of [16] prior to residual coding was also applied selectively. In addition to quantizing the residual, the prediction and transform parameters were counted towards the bitrate of each method as discussed in Sec. 2.4.

Compression method	Ratio
FLAC baseline	.540
Single-channel, Eq. 2	.540
multichannel, Eq. 3	.538
multichannel, Eq. 3 + SVD	**.505**

Table 1: Average compression ratios of various models (smaller is better) for 5.0 mixes across 100 music pieces.

As mentioned in Sec. 2.4, MPEG-ALS and TrueHD include a more involved multichannel prediction models. The reasons for not directly comparing against them here are the lack of availability of such software, and the related fact that FLAC is more widely adopted. However, some of our investigated methods are related. See Sec. 3.3 for further discussion.

3.3 Results

3.3.1 5.0 Music

Tables 1 and 2 show the average compression ratios for the 100 songs tested. Ratio per file was calculated as the total number of bits divided by original number of bits of the uncompressed 16-bit PCM representation.

To establish the baseline performance of the proposed methods in a traditional lossless coding situation, Table 1 compares coding the 5.0 mix alone. It can be seen that the single-channel model of Eq. 2 captures the compression efficiency of FLAC even with constant frame length and prediction order. Thus, modern 5.0 music mixes do not seem to benefit much from frame-based signal-adaptive options available in FLAC.

Introducing a more complex multichannel signal model of Eq. 3 for 5.0 content does not directly bring notable benefit compared to a single-channel model. However, introducing SVD as a post-processing step with Eq. 3 results in a notable improvement of compression ratio (.505 vs .540). This is taking into account the increased amount of side information that needs to be sent with SVD and multichannel prediction.

As discussed in Sec. 2.3, hierarchical systems rely on sending multiple mixes, in this case 2.0 and 5.0, simultaneously. Table 2 illustrates the results of such a scenario, where a stereo (2.0) downmix is utilized to reconstruct the 5.0 upmix. Since the 2.0-mix does not

Compression method	Ratio
FLAC baseline	.545
Single-channel + dmx, Eq. 4 + SVD	.527
multichannel + dmx, Eq. 5 + SVD	**.387**

Table 2: Average compression ratios of various models (smaller is better) across 100 music pieces, when coding both stereo (2.0) and a 5.0 mix of the same content simultaneously. The 2.0 is always sent with FLAC, reconstructed first, and utilized to reconstruct the 5.0 upmix. FLAC baseline codes all mixes separately.

use hierarchical prediction, nor was the use of complex non-hierarchical models found to benefit, it can be sent with established stereo coding, such as FLAC in this experiment.

As the benefit of SVD was established in Table 1, we wanted to next investigate it for the hierarchical methods. The single-channel method of Eq. 4 in combination with SVD seems to give adequate advantage compared to FLAC. The real benefit of using the downmix emerges when using the multichannel model of Eq. 5, especially when combined with the SVD projection. The benefit compared to the FLAC baseline is clear when calculating the compression ratio of sending both stereo (2.0) and a 5.0 version of the same content (.387 vs .545). This implies that global parameter optimization can be an important factor for the success of complex signal model predictors.

3.3.2 7.0.4 Immersive Audio

We first establish the performance baseline of the proposed methods for non-hierarchical lossless coding of the 7.0.4 mix alone. Table 3 shows a comparison to FLAC. Due to the limitation of FLAC to 8 channels, we separate the 7.0 - part and the 4 height channels to different files, and calculate the combined FLAC bitrate for both as the FLAC baseline. For this content, the benefit of our proposed models and SVD is not as good as for coding 5.0 material. In particular, the benefit of the multichannel model of Eq. 3 is not clear anymore, and the single-channel Eq. 2 is better. This indicates that these 7.0.4 mixes had less exploitable inter-channel correlation. However, the scope of our 7.1.4 database was much more limited than our 5.1 music database, which makes strong generalizations challenging.

Compression method	Ratio
FLAC baseline	.444
Single-channel, Eq. 2 + SVD	**.437**
multichannel, Eq. 3 + SVD	.443

Table 3: Average compression ratios of various models (smaller is better) for 7.0.4 mixes across 19 immersive cinematic and music clips.

Compression method	Ratio
FLAC baseline	.466
Single-channel + dmx, Eq. 4 + SVD	.450
multichannel + dmx, Eq. 5 + SVD	**.364**

Table 4: Average compression ratios of various models (smaller is better) across 19 immersive cinematic and music clips, when coding stereo (2.0), 5.0, and 7.0.4 mixes of the same content simultaneously. The 2.0 is always sent with FLAC, reconstructed first, and utilized to reconstruct the 5.0 upmix, which in turn is utilized to reconstruct 7.0.4. FLAC baseline codes all mixes separately.

Table 4 illustrates hierarchical coding of three simultaneous versions of the same content: stereo, 5.0, and 7.0.4, With our proposed hierarchical methods, stereo is again always sent with FLAC, and is used to construct 5.0, which in turn is used to construct the 7.0.4 mix. It can be again seen that especially the multichannel prediction model of Eq. 5 with SVD performs better than the baseline FLAC version of coding all mixes separately. This is in line with the results for the 5.0 music material in Table 2, and regardless of the non-hierarchical multichannel model not performing well for 7.0.4 coding alone. Although not detailed here, similar but less beneficial results were observed for cases of sending two mixes together with hierarchical coding (2.0 + 7.0.4 and 5.0 + 7.0.4). 5.0 mix was unsurprisingly noted to be a better predictor of a 7.0.4 mix than 2.0.

However, when analyzing a subset of the 7.1.4 immersive audio, it is clear that the cinematic material utilizing the EBU ADM object audio renderer to construct non-static mixes was more challenging for both standalone 7.0.4 coding, and hierarchical coding. Tables 5 and 6 show the results isolated for the cinematic part of the 7.1.4 database. The material was clearly more

Compression method	Ratio
FLAC baseline	.353
Single-channel, Eq. 2 + SVD	**.344**
multichannel, Eq. 3 + SVD	.357

Table 5: As in Table 3, but isolated across the 10 immersive cinematic clips that utilize the EBU ADM renderer [24] to generate non-static mixes.

Compression method	Ratio
FLAC baseline	.376
Single-channel + dmx, Eq. 4 + SVD	.375
multichannel + dmx, Eq. 5 + SVD	**.357**

Table 6: As in Table 4, but isolated across the 10 immersive cinematic clips that utilize the EBU ADM renderer [24] to generate non-static mixes.

sparse with a lower baseline FLAC compression ratio. Thus, there was arguably not much benefit to gain using more complex prediction or SVD. The static linear downmix utilized for the music material is not always the most realistic, and the use of EBU AMD renderer, possibly in combination with the sparsity, seems to result in more difficult material for the signal model. For a real-life codec, there would clearly be the need for adaptive switching and selection of the most suitable coding method per each frame, as now in FLAC.

3.3.3 Comparison to Prior Art

The MLP lossless compression system [8], and it's derivative TrueHD [9] implement a method for reducing channel correlations prior to using individual, cascaded transforms per channel. This is done via "affine transforms" that are not publicly detailed, but are possibly static and format-dependent (e.g. mid-side transform for stereo). These can be loosely interpreted as being related to using a transform like our SVD method in combination with the single-channel model of Eq. 2. The compression rates for 5.1-material reported in [8] seem notably worse than in our experiments even for the FLAC baseline, but the database was not detailed. Given the lack of details, we were not able to compare the SVD transform against MLP, other than to note that SVD is in theory optimal and adaptive to each frame signal. Also, in our preliminary tests we noted that using a SVD transform after prediction immediately prior to Rice coding produced better results than prior prediction, especially when using joint multichannel predictors (Eq. 3 and Eq. 5).

As mentioned in Sec. 1, MLP [8] also implements a version of hierarchical coding, which enables decoding a linear stereo downmix directly from a 6-channel bitsream without decoding the upmix first, and optionally decoding both stereo- and 5.1 mixes. This system relies on linear channel-combination downmixes, and the downmix coefficients being locally static in order to limit the amount of side information. In contrast, our proposed models are capable of non-constrained hierarchical modeling between arbitrary mixes. However, as seen in Sec. 3.3.2 when the relationships between the different mixes were non-static, our methods could potentially benefit from further research.

It should be noted that compression ratios better than ours (and the present FLAC result) were reported for MPEG-ALS with an unspecified 5.1 test set [6]. Since this occurred for a basic single-channel model as well, the content in [6] may have been sparser and more dynamic, with less surround- and center channel utilization (material being older), and the LFE channel being included. Most importantly, the MPEG-ALS use of cascaded multichannel prediction with cascaded Toeplitz systems did not have as significant a relative benefit compared to non-multichannel baseline, as the utilization of the downmix prediction, or SVD projection in this paper. Rather, it was comparable to the small difference between our single- and multichannel baseline models, Eq. 2 and Eq. 3.

We note that calculating the SVD per frame was the most notable source of increased computational complexity compared to traditional cascaded Toeplitz systems. In theory, SVD has complexity $O(L*C*min(L,C))$, where L is the number of samples in a frame, and C the number of channels. However, the added complexity is all in the encoder side, with decoding being limited to simple matrix operations. This is usually not an issue in codecs.

4 CONCLUSION

This work presents an evaluation of improved methods for the lossless compression of multichannel audio, both for the multichannel audio alone, and when the upmix is coded jointly together with a downmix, at the

cost of encoder computational complexity. Specifically, we propose using signal models not limited to traditional Toeplitz systems with a general linear solver, and SVD projection prior to Rice coding.

A 9% gain in compression ratio is achieved over FLAC, by utilizing a combination of multichannel prediction, SVD, and Rice coding, when sending 5.0 music content alone. For music material using linear ITU downmixes, the results show approximately a 30% improvement in the compression ratio, over FLAC, when both a stereo downmix and a 5.0 upmix are to be jointly encoded and hierarchically reconstructed. Similar benefit was observed for simultaneous sending and joint coding of stereo, 5.1, and 7.1.4 mixes of music material.

With 7.1.4 immersive cinematic material, although there was some advantage, the benefits of our proposed methods were not as notable likely due to the sparser nature of the material that was available. Also, the relation of a non-static downmix to the upmix seemed to be more difficult to model with our methods. Thus, further research on suitable signal models could be beneficial when the upmix and downmix relationship is more complex than the ITU downmix.

For material for which the signal models are valid, the proposed approaches could yield significant gains for data server storage and transmission of multichannel audio data. Implementation of frame-based method switching, silence handing, and other typical codec features may further improve results, as this enables adaption for different content types and varying material.

5 ACKNOWLEDGEMENTS

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

References

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

[2] 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.

[3] Zacharov, N., Holm Pedersen, T., and Pike, C., "A common lexicon for spatial sound quality assessment - latest developments," 2016.

[4] Riedmiller, J., Kjörling, K., Rödén, J., Wolters, M., Biswas, A., Boon, P., Carroll, T., Ekstrand, P., Gröschel, A., Hedelin, P., Hirvonen, T., Hörich, H., Klejsa, J., Koppens, J., Krauss, K., Lehtonen, H.-M., Linzmeier, K., Mehta, S., Muesch, H., Mundt, H., Norcross, S., Popp, J., Purnhagen, H., Resch, B., Samuelsson, J., Schug, M., Sehlström, L., Tsingos, N., Villemoes, L., and Vinton, M., "Delivering Scalable Audio Experiences using AC-4," *IEEE Transactions on Broadcasting*, 63(1), 2017.

[5] Poets, A., Preihs, S., and Peissig, J., "On the Evaluation of Perceived Spatial Immersion in the Application of Automatic Upmixing for 3D Surround Sound Systems," in *DAGA 2023 49th Annual Conference on Acoustics*, 2023.

[6] Liebchen, T., "MPEG-4 ALS - The Standard for Lossless Audio Coding," *The Journal of the Acoustical Society of Korea*, 28, 2009.

[7] Hellerud, E., Solvang, A., and Svensson, U. P., "Spatial redundancy in Higher Order Ambisonics and its use for lowdelay lossless compression," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009.

[8] Gerzon, M., Craven, P., Stuart, J. R., Law, M. J., and Wilson, R., "The MLP Lossless Compression System for PCM Audio," *Journal of Audio Engineering Society*, 52, 2004.

[9] Melkote, V., Law, M., and Wilson, R., "Hierarchical and Lossless Coding of audio objects in Dolby TrueHD," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[10] Van Beurden, M. and Weaver, A., "Free Lossless Audio Codec," Internet-draft, Internet Engineering Task Force, 2023,

https://datatracker.ietf.org/doc/draft-ietf-cellar-flac/10/.

[11] Robinson, T., "SHORTEN: lossless and near-lossless waveform compression," Technical report, Cambridge University Engineering Department, Cambridge, UK, 1994.

[12] Golomb, S. W., "Run-length encodings," *IEEE Transactions on Information Theory*, 12(3), 1966.

[13] Rice, R. F., "Some Practical Universal Noiseless Coding Techniques," in *JPL Publication 79-22, Jet Propulsion Laboratory*, 1979.

[14] Rissanen, J. and Langdon, G. G., "Arithmetic Coding," *IBM Journal of Research and Development*, 23(2), 1979.

[15] Briand, M., Martin, N., and Virette, D., "Parametric Representation of Multichannel Audio Based on Principal Component Analysis," in *Audio Engineering Society Convention 120*, 2006.

[16] Namazi, M., Elshafiy, A., and Rose, K., "Spatial Audio Compression with Adaptive Singular Value Decomposition Using Reconstructed Frames," in *Audio Engineering Society Conference: 2022 AES International Conference on Audio for Virtual and Augmented Reality*, 2022.

[17] Levinson, N., "The Wiener RMS error criterion in filter design and prediction." *J. Math. Phys.*, 25, 1975.

[18] Burg, J. P., *Maximum Entropy Spectral Analysis*, Ph.D. thesis, Stanford University, 1975.

[19] Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., and Sorensen, D., *LAPACK Users' Guide*, Society for Industrial and Applied Mathematics, third edition, 1999.

[20] Hilt, D. E. and Seegrist, D. W., *Ridge, a computer program for calculating ridge regression estimates*, volume 236, Upper Darby, Pa, Dept. of Agriculture, Forest Service, Northeastern Forest Experiment Station, 1977.

[21] ITU-R, "Audio Definition Model renderer for advanced sound systems," Recommendation BS.2127-0, International Telecommunication Union, Geneva, 2019.

[22] Netflix, "Netflix Open Source Content," 2022, https://opencontent.netflix.com/.

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

[24] ITU-BS.2076, "Audio Definition Model," Standard, International Telecommunication Union, Geneva, CH, 2019, https://github.com/ebu/ebu_adm_renderer.