This is a corrected version of the Author Accepted Manuscript submitted to ICASSP-2021
Please note that, as of 10 June 2021, the version on IEEE Xplore is incorrect despite best efforts to have it removed/replaced.

PROCESSING PIPELINES FOR EFFICIENT, PHYSICALLY-ACCURATE SIMULATION OF MICROPHONE ARRAY SIGNALS IN DYNAMIC SOUND SCENES

Alastair H. Moore[*†], *Rebecca R. Vos*[*], *Patrick A. Naylor*[*], *Mike Brookes*[*]

[*] Department of Electrical and Electronic Engineering, Imperial College, London, UK
[†] Square Set Sound, London, UK

ABSTRACT

Multichannel acoustic signal processing is predicated on the fact that the interchannel relationships between the received signals can be exploited to infer information about the acoustic scene. Recently there has been increasing interest in algorithms which are applicable in dynamic scenes, where the source(s) and/or microphone array may be moving. Simulating such scenes has particular challenges which are exacerbated when real-time, listener-in-the-loop evaluation of algorithms is required. This paper considers candidate pipelines for simulating the array response to a set of point/image sources in terms of their accuracy, scalability and continuity. A new approach, in which the filter kernels are obtained using principal component analysis from time-aligned impulse responses, is proposed. When the number of filter kernels is ≤ 40 the new approach achieves more accurate simulation than competing methods.

Index Terms— acoustic simulation, microphone arrays, head movement, hearing aids, virtual reality

1. INTRODUCTION

Microphone arrays are now routinely included in many devices including mobile phones, home voice assistants, mobile robots, augmented reality systems and hearing aids. Exploiting the direction-dependence of the interchannel relationships between the microphone signals, beamforming algorithms attenuate interfering sources and/or diffuse noise while preserving the wanted sound source. Recently, algorithms which account for [1, 2], or actively exploit [3], array motion have been proposed.

Development and evaluation of systems in this emerging field require the real-time simulation of complex dynamic scenarios in which the microphone signals are physically accurate, to the extent that they elicit reliable measures of algorithm performance. We concentrate specifically on the case of binaural hearing aids where the device at each ear typically contains two or three microphones and aspire to conduct listener-in-the-loop experiments where algorithms and acoustic scenarios can be varied systematically. Similar experiments using a master hearing aid (MHA) (e.g. [4]) have used a static lab environment [5] or virtual environments rendered through a loudspeaker array [6, 7]. In [8] a system which simulates microphone signals in real-time is presented however, at the time of publication, the update rate of early reflections was limited by computational resources, such that only the direct path signals could be considered physically accurate. It is, therefore, important to consider the most efficient means of simulating physically-accurate microphone signals in complex reverberant environments.

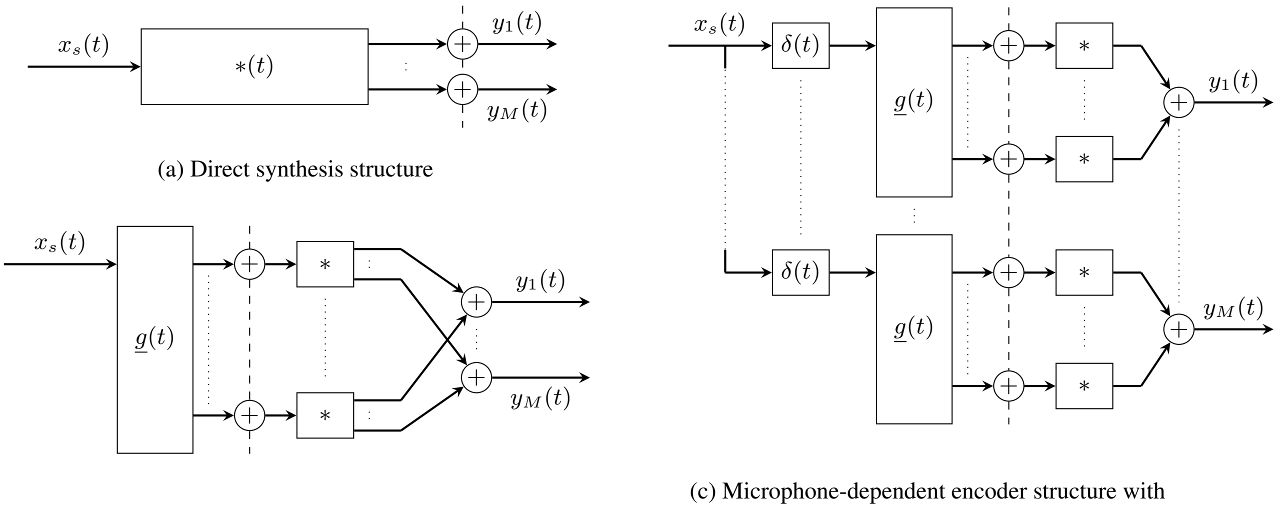
In this paper we are concerned with simulating the response of an array to a potentially large number of sources, representing the direct path and early reflections. We assume that the source locations are obtained using geometric acoustic methods [9, 10, 11, 12, 13]. In this approach, the sound field at the listener's head due to all the sources that are present is regarded as a superposition of plane waves arriving from different directions. The signal at each microphone may then be obtained by convolving each plane wave with the direction-dependent array manifold and summing over all waves. In a classic binaural system the array manifold is better known as the head-related impulse response (HRIR) whereas for hearing devices the term hearing aid head-related impulse response (HAHRIR) is becoming common. Since the time-of-flight of a particular ray is not, in general, an exact multiple of the sample period, it is necessary to interpolate each ray to account for the fractional sample delay [14] and ensure that inter-microphone delays are preserved.

Often, the array manifold must be interpolated from a limited number of measurement directions [15]. In [16] spatialization filters are obtained by interpolating between time-aligned measurements of the HRIR, with the inter-microphone time delay (ITD) reintroduced after convolution. The time alignment is needed to avoid comb-filtering effects. This is a time-aligned version of vector base amplitude panning (VBAP) [17] and we present a more efficient structure below in which delays are introduced before the convolution step. In [18] spherical harmonic (SH) interpolation of HRIRs was used to obtain time varying filters for dynamic spatialization. It was shown that more of the energy in the SH representation is concentrated at lower orders if the SH expansion is performed around microphone-centred co-ordinates. Recent investigations [19, 20, 21] have similarly concentrated the majority of the energy in fewer harmonics by time aligning HRIRs prior to SH expansion. Whilst a SH representation can be used to obtain HRIRs at interpolated directions, it is also possible to perform the convolution in the SH domain. In this case source signals are first encoded into a SH representation. Applying principal component analysis (PCA) directly to a database of HRIR for many individuals, [22] concluded that a good approximation to the original can be obtained even when as few as five of the principal components are retained. Treating each basis function as a filter kernel, spatialization can be obtained by encoding signals into the domain defined by the principal components.

In this paper we (i) determine the computational cost of alternative pipelines, (ii) investigate whether time-aligning the array manifold prior to performing PCA yields similar improvements as in SH and vector base amplitude panning (VBAP), (iii) investigate the number of filter kernels needed to obtain physically-accurate signals and (iv) examine the trade-off between accuracy and scalability.

This work was supported by the Engineering and Physical Sciences Research Council [grant number EP/S035842/1].

This is a corrected version of the Author Accepted Manuscript submitted to ICASSP-2021
Please note that, as of 10 June 2021, the version on IEEE Xplore is incorrect despite best efforts to have it removed/replaced.



(a) Direct synthesis structure

(b) Microphone-independent encoder structure

(c) Microphone-dependent encoder structure with time-alignment

Fig. 1: Pipeline structures for simulating microphone array signals showing signal flow for a single source where $*$, $\underline{g}(t)$ and $\delta(t)$ denote a filter operation, a vector multiplication, and a delay respectively. The vertical dashed line separates per-source, time-varying processing to the left from static processing blocks, which are common to multiple sources, to the right.

2. PROBLEM DEFINITION

In array-centered co-ordinates, the signal, $x_s(t)$, at the origin is the result of free-field propagation from a point source (or image source) at position $\mathbf{r}_s$. The resulting signal, $y_m(t)$, at the m-th microphone of an array is

$$y_m(t) = \int_\tau h_m(\Omega(t), \tau) x_s(t - \tau) d\tau \tag{1}$$

where $h_m(\Omega, t)$ is the array manifold, which depends on the direction of arrival, $\Omega(t)$, of the incident wave.

Movement of the source and/or the array appears as source movement in the array-centered co-ordinates. Assuming the source is in the far field, the propagation delay and attenuation effects of translation are encapsulated in $x_s(t)$. The effect of rotation is observed in the direction-dependence of $h_m(\Omega, t)$.

The goal of a simulation pipeline is to approximate, $y_m(t)$ at sample instants $t = n\Delta, n \in \mathbb{Z}$. In discrete time, (1) becomes

$$y_m(n\Delta) = \mathbf{h}_m^T(\Omega(n\Delta))\mathbf{x}_s(n\Delta) \tag{2}$$

where

$$\mathbf{h}_m(\Omega) = \begin{bmatrix} h_m(\Omega, 0) & h_m(\Omega, \Delta) & \dots & h_m(\Omega, (N-1)\Delta) \end{bmatrix}^T$$
$$\mathbf{x}_s(t) = \begin{bmatrix} x_s(t) & x_s(t - \Delta) & \dots & x_s(t - (N-1)\Delta) \end{bmatrix}^T.$$

To compare methods of simulating the time-varying filtering the following assumptions are made: (i) $h_m(\Omega, t)$ is known from critically sampled measurements in time and space, such that it can be interpolated to arbitrary angles over the frequencies of interest and expressed as a finite impulse response (FIR) filter of length N, (ii) processing is performed in blocks of length, L samples, (iii) $\Omega(t)$ is sampled once per block, (iv) a look-up table (LUT) which maps Ω to parameter/coefficient values can be accessed once per block with negligible computational cost, but not more often[1], and (v) time variation is implemented as linear interpolation between one or more values over the duration of one block using a per-sample increment — the increment is assumed to have negligible computational cost but there is an overhead of one multiply per interpolated parameter in each block.

Desirable properties of a processing pipeline are (i) scalability — maximising the number of sources which can be synthesised; (ii) numerical accuracy — signal processing methods require certain acoustic features to be fulfilled. In particular the relative transfer function (RTF) between microphones obtained from the simulated signals should be close to the true RTF for a plane wave; and (iii) perceptual fidelity — simulated signals should be free of glitches (e.g. discontinuities) and direction dependent spectral modifications (e.g. comb filtering).

3. PROCESSING PIPELINES

Potential processing pipelines are considered according to their overall structure, as shown in Fig. 1. For each, the computational cost is estimated according to the number of multiplication operations required under the assumptions outlined in Section 2. The computational cost per block of each pipeline is expressed in Table 2 as a function of the parameters listed in Table 1. The dependence of each pipeline on the number of microphones, M, and the number of sources, S is particularly important in choosing the most appropriate pipeline for a particular application. In all cases, F/L blocks must be processed each second.

3.1. Direct synthesis methods

Direct synthesis methods, depicted in Fig. 1(a), perform a filtering operation independently for each source. Time variation is implemented by changing the filter.

Interpolated FIR The structure of the processing follows (2). The LUT returns the required $\mathbf{h}_m(\Omega)$ and the N filter coefficients are incremented on each sample. Since the filter changes at each sam-

[1] The validity of this assumption depends on the size of cache available and the size of the LUT.

Symbol	Definition
F	sample rate
L	samples per buffer
S	number of point sources
M	number of microphones
J	number of non-zero kernel coefficients
K	number of kernels (filtering operations)
N	kernel length
D	implement delay (number of sinc coefficients)
T_f	forward FFT (N*log(N))
T_r	reverse FFT (N*log(N))
C	multiplication of signal with kernel (N/2)

Table 1: List of variables in computational cost

Pipeline	Computational cost
FIR	$MSN + MSLN$
Delay only	$\mathcal{D} = MS + MSLD$
Direct FFT	$ST_f + MSC + MT_r$
NSPK	$2SL + KT_f + \mathcal{F}$
VBAP	$S(2J) + S(2J)L + KT_f$
SH	$SK + SKL + KT_f + \mathcal{F}$
GDA VBAP	$\mathcal{D} + S(2J) + M(S(2J)L + KT_f) + \mathcal{F}$
GDA SH	$\mathcal{D} + SK + M(SKL + KT_f) + \mathcal{F}$
PCA	$M\,(SK + SKL + KT_f) + \mathcal{F}$
GDA PCA	$\mathcal{D} + M(SK + SKL + KT_f) + \mathcal{F}$

Table 2: Computational cost of pipelines where $\mathcal{F} = MKC + MT_r$

ple, there is no advantage to frequency domain convolution so direct convolution is used with LN multiplies per source, per microphone.

Delay only For an ideal free-field array $\mathbf{h}_m(\Omega)$, reduces to a direction-dependent fractional delay [14]. The per-microphone delay is incremented on each sample and sinc interpolation is implemented using a precomputed, oversampled sinc wavetable. The computational cost depends on the order of the sinc interpolation, with the number of coefficients here denoted as D.

Direct FFT The array manifold is fixed for the duration of each block, rather than interpolated. The LUT returns the array manifold, pre-transformed into the frequency domain. One forward transform, with cost T_f, is required per source and one reverse transform, with cost T_r, per microphone. The cost of a fast Fourier transform (FFT) is hardware dependent but $N \log_2(N)$ is a reasonable approximation. The cost of complex multiplication depends on the hardware and is denoted C.

3.2. Microphone-independent encoder methods

Microphone-independent encoder methods, depicted in Fig. 1(b), use a time-varying gain to assign each source signal to one or more of K busses. Each bus is filtered by a fixed filter kernel per microphone and microphone signals are obtained by summing over the corresponding filter outputs. Taken together the kernels impart the required direction-dependent filtering, including interchannel phase differences. To avoid transients, the encoding gains are interpolated from the previous to the new vales over the duration of the frame.

Virtual loudspeakers (NSPK) Filter kernels are the array manifold corresponding to a grid of directions. The LUT returns the index of the nearest loudspeaker and the source is assigned to the bus for that loudspeaker. Using the nearest loudspeaker ensures the microphone signals correspond to a valid, but likely wrong, direction. Interpolation between previous and new loudspeakers is achieved using fixed fade out/in at a cost of two multiplies per sample.

Virtual speakers (VBAP) A weighted sum of the source signal is sent to neighbouring loudspeakers where J is 2 for loudspeakers on a circle or 3 for a spherical distribution [17]. The LUT returns the panning weights. The number of active speakers per source per block is upper bounded by $2J$. If the direction of arrival (DOA) changes rapidly there may be intermediate speakers which are unused.

Spherical harmonics (SH) The kernels are the spherical Fourier transform (SFT) [23] of $\mathbf{h}_m(\Omega, t)$ with one kernel per SH component. For a transform of order Q, $K = (Q+1)^2$, or if Ω is constrained to the horizontal plane, $K = 2Q+1$. The LUT returns the precomputed result of evaluating the SH function for the required DOA. Regardless of the DOA all sources are fed to all kernels.

3.3. Time-aligned encoder methods

It has been proposed that time-aligning the array manifold, such that the onset time is independent of direction, can improve the accuracy of interpolation. This requires that direction-dependent time delays are applied separately, as shown in Fig. 1(c). Compared to the encoder methods in Section 3.2, the additional cost of implementing this delay is the same as implementing the 'Delay only' pipeline. Furthermore, since the inputs to each kernel are decoupled, the forward FFTs must be computed independently for each microphone.

In this work we use the negative-sloped zero crossing of the energy weighted group delay to estimate the onset times, as in [24], and refer to the methods as group delay-aligned (GDA).

GDA VBAP Time-alignment avoids the introduction of comb-filtering artifacts when the same signal is presented from different speakers arriving with different delays.

GDA SH Time-alignment is applied before performing the SFT which reduces the spatial bandwidth associated with the direction-dependence of the onset.

3.4. Principal component analysis (PCA) methods

Using basis functions obtained from PCA as the filter kernels ensures that the maximum amount of variance in the array manifold is accounted for in fewest number of kernels. The encoder weights returned by the LUT are data driven.

PCA The encoding weights are microphone dependent and so all processing scales with M.

GDA PCA It is proposed that time-alignment of the array manifold prior to computing the principal components will allow a better reconstruction of the array manifold using fewer kernels.

4. EVALUATION

To evaluate the physical accuracy of the rendering pipelines at each microphone individually, simulated impulse responses, $\hat{h}_m(\Omega, t)$, created using each of the methods were compared to ground truth impulse responses, $h_m(\Omega, t)$, for sources on a horizontal grid with 1° resolution. The mean and worst case root mean squared (RMS)

This is a corrected version of the Author Accepted Manuscript submitted to ICASSP-2021
Please note that, as of 10 June 2021, the version on IEEE Xplore is incorrect despite best efforts to have it removed/replaced.

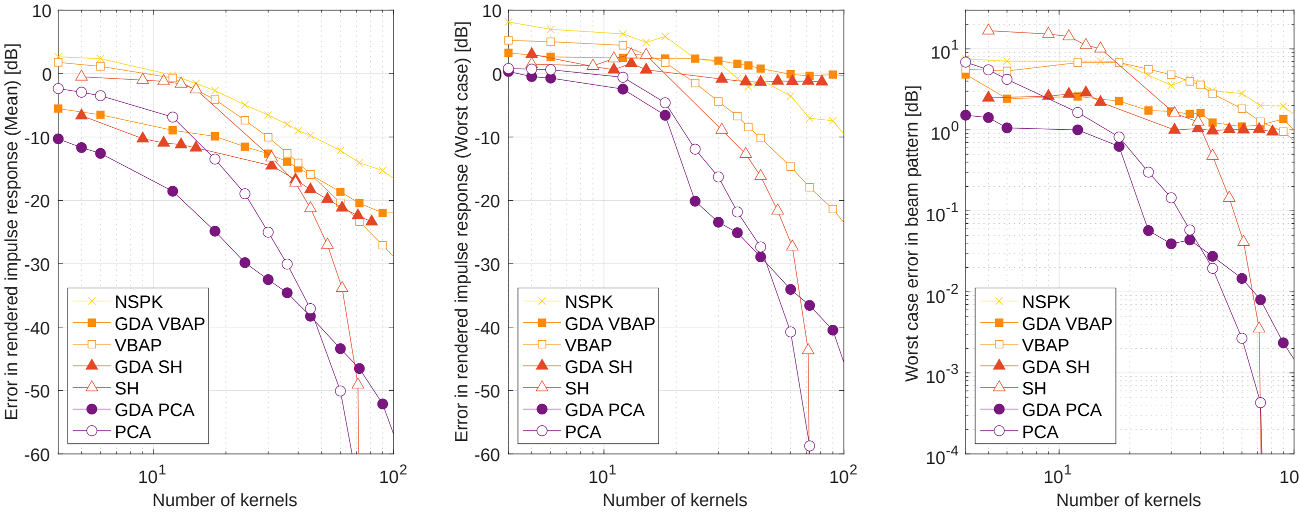


Fig. 2: Accuracy of rendering pipelines as a function of the number of kernels per microphone. (left) mean error over direction, (middle) error at worst case direction, (right) worst case error in effective beampattern.

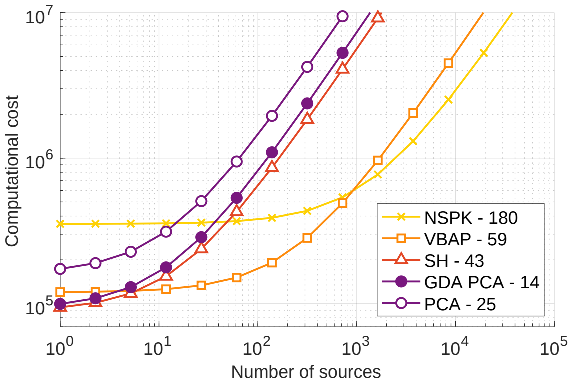


Fig. 3: Computational cost as a function of S. For each pipeline the number of kernels, K, indicated in the legend, is the minimum required to achieve mean error of −20 dB (see left panel of Fig. 2). $F = 48000$, $L = 128$, $M = 4$, $N = 256$, $J = 2$, $D = 0$.

error over all directions is reported for each pipeline. The mean error gives a consistent impression of the overall accuracy whereas the worst case error is more appropriate for defining the limiting case of acceptability for a particular task. Of particular interest for array signal processing is whether the simulated microphone signals produce the expected response at the output of a beamformer. A minimum variance distortionless response (MVDR) beamformer is designed for isotropic noise using the ground truth array manifold. The power of the beamformer output in response to the ground truth array manifold is the target beampattern whereas the effective beampattern is the power in response to the simulation pipelines. The worst case error in the beampattern is the difference between the target and effective beampattern over Ω.

Fig. 2 shows the accuracy of each pipeline as a function of the number of kernels, K. For each metric, when a relatively small number of kernels is used, the time-aligned methods outperform their conventional counterparts. This is consistent with previously reported results [18, 20]. As more components are used the conventional methods perform better whereas the worst case error for GDA VBAP and GDA SH plateaus. For very good accuracy (≤−40 dB error in the rendered impulse response or ≤−0.01 dB error in the beampattern) the SH and PCA methods require the fewest kernels.

To compare across all methods the number of kernels is only one factor in the computational cost. Choosing a mean error in the rendered impulse response of −20 dB as an obtainable level of accuracy for all methods, the scalability of each pipeline is shown in Fig. 3, with the values of the variables given in the caption. Note that $D = 0$, which effectively makes the cost of implementing the fractional delay free, is chosen to highlight the additional cost of individually encoding and performing the forward transform for each microphone. Each pipeline has a trade-off between the overhead in computing the kernel compared to the cost of encoding one extra source. GDA PCA requires the fewest kernels and yet, even neglecting the cost of implementing the per-source, per-microphone delay, it is more expensive than SH. Of the microphone-independent encoder methods, SH requires the fewest kernels and has the lowest overhead. However, for $S > 5$ it can be seen that other methods require less computation. VBAP is optimal for $5 < S < 90$ with NSPK offering the best scalability for $S > 90$. It should be noted that Fig. 3 relates to one specific level of accuracy, for which NSPK was the limiting factor, and one specific array.

5. DISCUSSION AND CONCLUSIONS

The computational cost and accuracy of a range of simulation pipelines have been presented. Of particular interest is a recently proposed class of pipelines in which the filter kernels are obtained by first time-aligning the array manifold. Of these, the GDA PCA pipeline was the most accurate in our tests. However, in the case studied here, the cost of encoding and transforming each source signal separately for each microphone outweighed the saving gained from using fewer kernels.

This is a corrected version of the Author Accepted Manuscript submitted to ICASSP-2021
Please note that, as of 10 June 2021, the version on IEEE Xplore is incorrect despite best efforts to have it removed/replaced.

6. REFERENCES

[1] M. Zohourian, A. Archer-Boyd, and R. Martin, “Multi-channel speaker localization and separation using a model-based GSC and an inertial measurement unit,” in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, 2015, pp. 5615–5619.

[2] A. H. Moore, L. Lightburn, W. Xue, P. A. Naylor, and M. Brookes, “Binaural mask-informed speech enhancement for hearing aids with head tracking,” in *Proc. Int. Workshop on Acoust. Signal Enhancement (IWAENC)*, Tokyo, Japan, Sept. 2018.

[3] A. H. Moore, W. Xue, P. A. Naylor, and M. Brookes, “Noise covariance matrix estimation for rotating microphone arrays,” *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 27, no. 3, pp. 519–530, Mar. 2019.

[4] G. Grimm, T. Herzke, D. Berg, and V. Hohmann, “The master hearing aid: A PC-based platform for algorithm development and evaluation,” *Acta Acustica united with Acustica*, vol. 92, no. 4, pp. 618–628, July 2006.

[5] B. Cornelis, M. Moonen, and J. Wouters, “Speech intelligibility improvements with hearing aids using bilateral and binaural adaptive multichannel wiener filtering based noise reduction,” *J. Acoust. Soc. Am.*, vol. 131, no. 6, pp. 4743–4755, 2012.

[6] G. Grimm, M. M. Hendrikse, and V. Hohmann, “Interactive rendering of dynamic virtual audio-visual environments for “subject-in-the-loop” experiments,” *J. Acoust. Soc. Am.*, vol. 146, no. 4, pp. 2801–2801, Oct. 2019.

[7] M. M. E. Hendrikse, G. Llorach, V. Hohmann, and G. Grimm, “Movement and gaze behavior in virtual audiovisual listening environments resembling everyday life,” *Trends Hear.*, vol. 23, pp. 1–29, Jan. 2019.

[8] F. Pausch, L. Aspöck, M. Vorländer, and J. Fels, “An extended binaural real-time auralization system with an interface to research hearing aids for experiments on subjects with hearing loss,” *Trends Hear.*, vol. 22, pp. 1–32, Oct. 2018.

[9] J. B. Allen and D. A. Berkley, “Image method for efficiently simulating small-room acoustics,” *J. Acoust. Soc. Am.*, vol. 65, no. 4, pp. 943–950, Apr. 1979.

[10] J. Borish, “Extension of the image model to arbitrary polyhedra,” *J. Acoust. Soc. Am.*, vol. 75, no. 6, pp. 1827–1836, June 1984.

[11] A. Krokstad, S. Strom, and S. Sørsdal, “Calculating the acoustical room response by the use of a ray tracing technique,” *J. of Sound and Vibration*, vol. 8, no. 1, pp. 118–125, July 1968.

[12] A. Kulowski, “Algorithmic representation of the ray tracing technique,” *Applied Acoust.*, vol. 18, no. 6, pp. 449–469, 1985.

[13] A. Wabnitz, N. Epain, C. Jin, and A. van Schaik, “Room acoustics simulation for multichannel microphone arrays,” in *Proc. Int. Symp. on Room Acoust. (ISRA)*, 2010.

[14] P. M. Peterson, “Simulating the response of multiple microphones to a single acoustic source in a reverberant room,” *J. Acoust. Soc. Am.*, vol. 80, no. 5, pp. 1527–1529, Nov. 1986.

[15] K. Hartung, J. Braasch, and S. J. Sterbing, “Comparison of different methods for the interpolation of head-related transfer functions,” in *Proc. Audio Eng. Soc. (AES) Conf. on Spatial Sound Reproduction*, Rovaniemi, Finland, Apr. 1999.

[16] M. Cuevas-Rodríguez, L. Picinali, D. González-Toledo, C. Garre, E. Rubia-Cuestas, L. Molina-Tanco, and A. Reyes-Lecuona, “3D Tune-In Toolkit: An open-source library for real-time binaural spatialisation,” *PLOS ONE*, vol. 14, no. 3, pp. e0211899, Mar. 2019.

[17] V. Pulkki, “Uniform spreading of amplitude panned virtual sources,” in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, Oct. 1999, pp. 187–190.

[18] J.-G. Richter, M. Pollow, F. Wefers, and J. Fels, “Spherical harmonics based HRTF datasets: Design, implementation and evaluation for real-time auralization,” in *Proc. Int. Conf. on Acoust. (DAGA)*, Merano, Mar. 2013.

[19] H. Ziegelwanger and P. Majdak, “Modeling the direction-continuous time-of-arrival in head-related transfer functions,” *J. Acoust. Soc. Am.*, vol. 135, no. 3, pp. 1278–1293, Mar. 2014.

[20] C. Pike and A. I. Tew, “Subjective Assessment of HRTF Interpolation with Spherical Harmonics,” in *Proc. Int. Conf. on Spatial Audio (ICSA)*, Graz, Austria, Sept. 2017.

[21] F. Brinkmann and S. Weinzierl, “Comparison of head-related transfer functions pre-processing techniques for spherical harmonics decomposition,” in *Proc. Audio Eng. Soc. (AES) Conf. on Audio for Virtual & Augmented Reality*, Redmond, Aug. 2018, Audio Engineering Society.

[22] D. J. Kistler and F. L. Wightman, “A model of head-related transfer functions based on principal components analysis and minimum-phase reconstruction,” *J. Acoust. Soc. Am.*, vol. 91, no. 3, pp. 1637–1647, Mar. 1992.

[23] B. Rafaely, *Fundamentals of Spherical Array Processing*, Springer Topics in Signal Processing. Springer, 2015.

[24] A. Kounoudes, P. A. Naylor, and M. Brookes, “The DYPSA algorithm for estimation of glottal closure instants in voiced speech,” in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Orlando, May 2002, pp. 349–352.