Efficient Construction of the Spatial Room Impulse Response

Carl Schissler* Peter Stirling† Ravish Mehra‡

Oculus & Facebook

Figure 1: Our perceptually-based spatial room impulse response construction technique can efficiently generate spatial audio for sound propagation with low latency in interactive scenes: (left) Apartment; (center) City; (right) Temple.

Abstract

An important component of the modeling of sound propagation for virtual reality (VR) is the spatialization of the room impulse response (RIR) for directional listeners. This involves convolution of the listener's head-related transfer function (HRTF) with the RIR to generate a spatial room impulse response (SRIR) which can be used to auralize the sound entering the listener's ear canals. Previous approaches tend to evaluate the HRTF for each sound propagation path, though this is too slow for interactive VR latency requirements. We present a new technique for computation of the SRIR that performs the convolution with the HRTF in the spherical harmonic (SH) domain for RIR partitions of a fixed length. The main contribution is a novel perceptually-driven metric that adaptively determines the lowest SH order required for each partition to result in no perceptible error in the SRIR. By using lower SH order for some partitions, our technique saves a significant amount of computation and is almost an order of magnitude faster than the previous approach. We compared the subjective impact of this new method to the previous one and observe a strong scene-dependent preference for our technique. As a result, our method is the first that can compute high-quality spatial sound for the entire impulse response fast enough to meet the audio latency requirements of interactive virtual reality applications.

Keywords: Spatial audio, HRTF, sound propagation, spherical harmonics

Index Terms: H.5.5 [Information Interfaces and Presentation]: Sound and Music Computing—Signal analysis, synthesis, and processing; I.3.8 [Computer Graphics]: Applications

1 Introduction

A new generation of consumer head-mounted displays has encouraged the development of technologies that enhance the multi-modal virtual reality experience. An often-overlooked component of virtual reality (VR) is the rendering of realistic sound effects that correspond closely to the user's expectations. If the audio in a virtual environment is poorly rendered, it can interfere with the user's sense of immersion and presence [13]. On the other hand, high-quality sound can enhance immersion, source localization, and other subjective criteria [4].

The auralization of sound in a virtual environment involves several components, including the modeling of sound sources, listeners, and the effects of sound propagation within the environment. Sound propagation, arguably the most complex to simulate, requires the computation of phenomena like high-order reflections, diffraction, and scattering which are produced by interaction of sound with the scene. The effects of sound propagation are described by a filter called the *room impulse response* (RIR) that represents the transfer function between a particular source and listener pair within the environment. For an omnidirectional point source and listener, the RIR is a function of time and frequency. For directional sources and listeners, the RIR is also a function of the direction of sound emission from the source as well as the direction of sound arrival at the listener. The RIR is frequently used in room acoustic simulations for architectural applications where the distribution and decay of sound energy is an important quantity. However, the RIR cannot directly be used for auralization in VR because it does not incorporate the directional effects of the listener's head.

The modeling of listener directivity is frequently referred to as *spatial sound*. The goal of spatial sound is to reproduce the differences in sound heard at each ear by filtering the left and right channels according to the direction of sound arrival. This gives the user the sensation that the sound source is localized at a particular position in 3D space. In order to render sound propagation effects with a directional listener, the spatial sound filter for the listener must be applied to the RIR in order to generate a *spatial room impulse response* (SRIR). The SRIR consists of a separate time-domain filter for the left and right channels, and it contains the effect of both the environment and the listener's head on the sound emitted by the source. To render the sound heard at the listeners position, the anechoic audio from the sound source can be convolved with the SRIR and then played to the user over headphones.

A significant challenge for virtual reality applications is that the SRIR must be updated at a rate that is fast enough for the user to notice no perceptible latency. A commonly used threshold for the maximum end-to-end system latency is roughly 100ms [25]. This means that the total time it takes to recompute the RIR, apply spatial

*E-mail: carl.schissler@gmail.com
†E-mail: peter.stirling@oculus.com
‡E-mail: ravish.mehra@oculus.com

sound to generate a SRIR, interpolate the convolution system to the new SRIR, and reproduce the audio through headphones must be less than the maximum latency. If not, sound can seem to lag behind the user's current head position. While much work has been done to reduce the latency of sound propagation for interactive applications, previous techniques for generation of the SRIR from the resulting RIR may take over 500ms or more for a single sound source and therefore are too slow to meet this latency target.

In this work, we present a technique for the computation of the SRIR that is nearly an order of magnitude faster than previous approaches. Our method uses a spherical harmonic (SH) basis representation of the spatial sound field. While the use of spherical harmonics for spatial sound rendering is not a new idea, we introduce a novel perceptual metric based on the threshold of hearing which is used to evaluate the directivity strength at different parts of the RIR. For parts of the RIR with weak directivity, we use a lower-quality spherical harmonic representation of the spatial sound that is chosen by our metric to be perceptually similar to the full representation. Since the lower-quality spatial sound representation is much more efficient to compute, this approach enables the SRIR to be constructed quickly without affecting the perceptible quality of the sound.

We have evaluated our approach on a wide range of indoor and outdoor virtual environments and observe that our sound propagation and rendering system is capable of meeting a maximum latency target of 100ms and is $6.7-9.1$ times faster than the previous state of the art. To evaluate the perceptual impacts of our method, we have also conducted a preliminary user study that shows a strong preference for our method when compared to the much slower state of the art. When 10x as much compute resources are used for the previous method, there is similar preference for both, indicating the differences between our technique and the previous one are negligible when the latency is the same.

2 Background

2.1 Sound Propagation

The simulation of sound propagation is a well-studied problem with many applications in architectural acoustics, games, and virtual reality. The most accurate techniques are those based on numerically solving the Helmholtz wave equation. These include time-domain methods like finite-difference time domain [26] and adaptive rectangular decomposition [22] where a pressure field is evolved over time, or frequency-domain methods such as the boundary-element method [12] or the equivalent source method [17] which compute the pressure response for each frequency. While accurate, wave-based sound propagation algorithms scale very poorly when applied to high frequencies and the large, complex, and dynamic environments found in interactive VR experiences. As a result, these methods are usually only used for low frequencies in precomputed static environments with static source locations.

Another class of sound propagation techniques are based on the assumption that the primitives in the scene are much larger than the wavelengths of sound under consideration. These are the so-called *geometric acoustics* algorithms and they tend to be much faster to compute but also less accurate at low frequencies since they do not directly handle diffraction effects. Wave effects must be modeled separately using algorithms such as the uniform theory of diffraction (UTD) [33] or the Biot-Tolstoy-Medwin formulation [31] which approximate wave propagation over an edge at low frequencies. Many algorithms have also been proposed for the modeling of specular early reflections. These include the image source method [2, 5], beam tracing [10], and frustum tracing [6]. However, these approaches cannot model sound diffusion or scattering from rough surfaces and become slow for high-order reflections. Ray tracing or sound particle algorithms which use Monte Carlo integration to numerically solve the sound transport problem are frequently used for the case of diffuse reflections [8, 7]. In these methods, many rays or particles are emitted from a source in all directions and randomly scattered through the scene until a path to the listener is found. Various hybrid approaches have also been proposed that use some combination of the image source method and high-order diffuse path tracing [34, 15, 29]. Acoustic radiosity methods can also handle diffuse reflections but require long pre-computation time [19]. A recent area of research is the use of temporal coherence in the sound field to accelerate the computation of Monte Carlo ray tracing algorithms [29, 27]. In these approaches, a cache of previous sound propagation results is used to reduce the number of rays that are required on each simulation time step, thereby improving the overall system latency.

2.2 Spatial Sound

In spatial sound, the goal is to model the impact that the listener's head, ear, and torso geometry has on the sound arriving at the entrance of each ear. By filtering each sound arrival according to its direction, spatial sound algorithms emulate phenomena like inter-aural level differences, inter-aural time differences, and spectral differences between the ears. Accurate spatial sound rendering gives the user the impression that a sound source is localized at a particular position in 3D space.

Amplitude Panning: The most straightforward methods for spatial sound rendering are based on modeling only inter-aural level differences and fall into the category of *amplitude panning*. In these approaches, a gain coefficient for each channel is computed according to the angles between the sound arrival direction and the speaker positions. The result is that the sound source is localized between the speakers that are closest to the sound source. Vector-based amplitude panning (VBAP) is the most commonly used technique and it handles panning among arbitrary 2D or 3D speaker arrays [20]. While very efficient to compute, a significant drawback of amplitude panning is that it has limited spatial resolution and does not perform well with just two channels because there is no way to disambiguate the front-back or vertical position of the source.

Head-Related Transfer Functions (HRTF): The complex scattering effects due to the listener's head and torso can be described by a filter on the spherical domain called the *head-related transfer function* (HRTF). The HRTF, $H(\mathbf{x},t)$, is a function of Cartesian direction $\mathbf{x}$ and time t. It specifies an audio filter for the left and right ears for every direction relative to the listener. Most frequently, the HRTF is measured for a specific individual over a spherical grid of discrete directions in an anechoic chamber. A sound source can be localized in a particular direction $\mathbf{x}$ by interpolating the nearest measured left and right channel filters, $H_L(\mathbf{x},t)$ and $H_R(\mathbf{x},t)$, then convolving the filter for each channel with the source's audio. If the resulting audio is reproduced to the user over headphones, the sound source will seem to be localized in direction $\mathbf{x}$. For clarity, we drop the left/right subscripts for the remainder of this work and assume the computation can be performed the same for each channel.

2.3 Spherical Harmonics and Spatial Sound

The spherical harmonics (SH) are a set of orthogonal basis functions for the spherical domain and are denoted by $Y_{lm}(\mathbf{x})$, where $\mathbf{x}$ is a unit-length Cartesian direction, $l = 0, 1, ...n$ and $m = -l, ..., 0, ...l$. n represents the maximum spherical harmonic order. For order n, there are $(n+1)^2$ basis functions. An arbitrary spherical function $f(\mathbf{x})$ can be projected into the SH basis by evaluating an integral over the spherical domain to generate SH coefficients f_{lm}:

$$f_{lm} = \iint_{\mathbb{S}} Y_{lm}(\mathbf{x}) f(\mathbf{x}) d\mathbb{S}. \tag{1}$$

This integral can be evaluated using the discrete spherical harmonic transform or Monte Carlo numerical integration [35, 21]. With

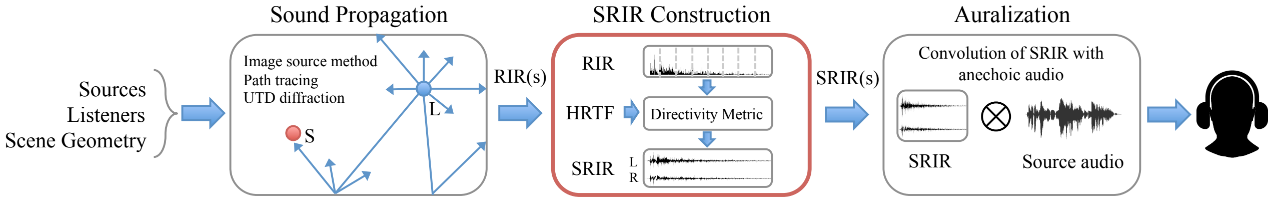


Figure 2: Overview of our spatial sound propagation and rendering pipeline. Given sound source(s), listener(s), and the scene geometry, the sound propagation module computes a room impulse response (RIR) for each source and listener pair. Then, the RIR(s) are converted to spatial room impulse responses (SRIR) using our efficient technique, highlighted in red, that uses a perceptual directivity metric to adaptively choose the spherical harmonic order for the HRTF at each part of the IR. Finally, the SRIR(s) are convolved with the corresponding source audio streams and presented to the user via headphones.

the Monte Carlo method, the SH coefficients are computed as a weighted sum of basis functions evaluated at a set of N uniformly-distributed random samples $\mathbf{x}_i$:

$$f_{lm} = \frac{1}{\sum_{i=0}^{N} f(\mathbf{x}_i)} \sum_{i=0}^{N} Y_{lm}(\mathbf{x}_i) f(\mathbf{x}_i). \tag{2}$$

Once the function is transformed into the SH basis, an approximation of the function, $\tilde{f}(\mathbf{x})$, can be computed in any direction $\mathbf{x}$:

$$\tilde{f}(\mathbf{x}) = \sum_{l=0}^{n} \sum_{m=-l}^{l} Y_{lm}(\mathbf{x}) f_{lm}. \tag{3}$$

If this process is applied to the HRTF, the result for each ear is a set of SH coefficients $h_{lm}(t)$. Due to the orthogonality of the spherical harmonics, if the sound arriving at the listener at a time sample from all directions is expressed in SH coefficients X_{lm}, then the HRTF for that time sample can be efficiently computed using a dot product of the basis function coefficients:

$$\tilde{H}(t) = \sum_{l=0}^{n} \sum_{m=-l}^{l} X_{lm} h_{lm}(t) \tag{4}$$

This property of the spherical harmonics is important for the efficient application of the HRTF to the room impulse response.

2.4 Spatial Room Impulse Response (SRIR) Construction

When sound propagation is simulated within an environment, the output at each simulation step is called a *room impulse response* (RIR). The RIR contains only the effect of the environment on the sound heard at the listener and can be represented in a few different ways. Wave-based sound propagation systems usually compute the RIR as an array of time-domain pressure samples, $p(t)$. The monaural sound heard by the listener can be directly obtained by convolving $p(t)$ with the source's anechoic audio. To support directional listeners for wave-based sound propagation, the plane-wave decomposition of the pressure field can be used to spatialize the pressure impulse response [16].

On the other hand, geometric sound propagation systems usually compute the RIR in the sound intensity or sound energy domain for octave frequency bands, rather than directly in the pressure domain. In this case, the RIR can be represented as a list of N *sound paths* which correspond to the reflection or diffraction paths detected on the current frame via ray tracing. The jth path contains the following information:

- $I_{j,b}$ - the sound intensity for the jth path and bth sound propagation simulation frequency band.
- t_j - the time of arrival or delay time for the path.
- $\mathbf{x}_j$ - the Cartesian 3D direction from the listener's position in the direction of sound arrival.

We use this representation in our new spatial sound approach.

The canonical way to incorporate the HRTF into the spatial room impulse response is to interpolate the HRTF filter for each sound path direction in the RIR and then multiply by the pressure magnitude for the path [14]. We refer to this as the *per path* SRIR construction method. The pressure SRIR can be computed for frequency band b according to the relation,

$$p_b(t) = \sum_{j=0}^{N} H(\mathbf{x}_j, t) \otimes \delta(t - t_j) \sqrt{I_{j,b} z_0} \tag{5}$$

where z_0 is the characteristic specific acoustic impedance of the propagation medium. This is essentially a direct time-domain convolution of the HRTF with the RIR. To compute the final spatial pressure IR containing all frequency and direction-dependent sound propagation effects, the IRs for all simulation frequency bands must be band-pass filtered into their corresponding frequency bands and then summed:

$$p(t) = \sum_{b} BandPass_b(p_b(t)). \tag{6}$$

Alternatively, the HRTF can be filtered into separate frequency bands $H_b(\mathbf{x}, t)$ in a preprocessing step to eliminate the need for filtering at runtime.

This generates an SRIR that can be convolved with the anechoic source audio to produce the sound heard by the listener at its current position and orientation. A significant drawback of this method of SRIR generation is that the HRTF must be interpolated for every sound path, and the number of paths can be more than 10^5. It can take over 500ms to compute the SRIR for a single sound source in our optimized implementation. As a result, this technique is not suitable for interactive applications. It is also possible to cluster paths based on their direction to reduce the number of interpolations [15], but this reduces the quality and resolution of the spatial sound and is still too slow to meet the 100ms latency target for long impulse responses. An alternative approach that is commonly used in interactive auralization systems to save computation is to spatialize only the direct sound or early reflections with the HRTF, while the remainder of the RIR uses amplitude panning. However, this results in late reverberation that is less spacious due to the lack of frequency-domain filtering and interaural time differences. It can also be difficult to closely match the timbre of the HRTF and panning parts of the IR.

3 Overview

In this section we provide an overview of our spatial sound rendering pipeline. The components of our system are shown in Figure 2 and include sound propagation, spatial room impulse response (SRIR) construction, and auralization.

Sound Propagation: The input to the sound propagation module is a collection of sound source(s), listener(s), and the scene geometry with acoustic material properties specified per-triangle. The sound propagation module uses these data to compute a room impulse response (RIR) for each sound source and listener pair in the energy-based path representation discussed in section 2.4. We use a hybrid of the image source method and diffuse path tracing starting from the listener [28], combined with UTD diffraction. However, any geometric sound propagation algorithm with a similar RIR output format can also be used.

Spatial Room Impulse Response Construction: The next stage uses the RIR produced by the sound propagation module and the user's HRTF to generate a spatial room impulse response that incorporates the listener's current head orientation in relation to the environment. The main contribution of this work is a novel efficient perceptually-based algorithm that can compute a high-quality SRIR with low latency relative to previous approaches. Our approach uses the user's HRTF and threshold of hearing to choose the appropriate spherical harmonic order to use for the HRTF at each part of the impulse response. This is discussed in detail in section 4.

Auralization: In the final stage of the pipeline, the SRIR(s) generated in the previous stage are convolved with the anechoic audio streams for the corresponding sound source(s). The resulting spatial sound is presented to the user through headphones.

4 Efficient SRIR Construction

The efficient computation of the spatial room impulse response (SRIR) is a challenging problem when generating sound for interactive virtual reality. To overcome this obstacle, we present a novel technique for computation of the SRIR that is about an order of magnitude faster than previous approaches.

In Figure 3, we summarize our algorithm. The input to our approach is a room impulse response (RIR) that has previously been computed using a geometric sound propagation system. The sound paths in the IR are sorted into partitions of length L, and for each partition we evaluate the directivity strength using a perceptual metric based on the user's threshold of hearing and a spherical harmonic representation of the HRTF. Our metric adaptively determines the minimum spherical harmonic order $\tilde{n}$ that is required to represent the partition's spatial sound with no perceptible loss in quality. Then, we efficiently convolve the HRTF and the RIR partition in the spherical harmonic domain up to order $\tilde{n}$ to generate the SRIR for the partition. Finally, this partition SRIR is overlap-added at the corresponding position in the output SRIR. When all partitions have been processed, the result is a spatial pressure impulse response that can be convolved with anechoic source audio to render the sound at the listener's position.

4.1 Perceptual Directivity Metric

A main component of our approach is a novel metric that evaluates the minimum required spherical harmonic order $\tilde{n}$ for each partition in the room impulse response. Our metric works by examining the spatial distribution of sound energy arriving at the listener during the partition. If there is strong directional information in the partition, then a higher SH order will be required to accurately represent the sound field. Otherwise, if the partition is more diffuse, a lower SH order can be used that requires less computation. In most indoor environments, earlier partitions will tend to be more directional, while the later ones will be more diffuse. Our metric takes advantage of this property of the IR so that the expensive high-order HRTF is used only where necessary. A key feature of our metric is that it can be evaluated very efficiently, so that the time saved by using a low-order HRTF for some partitions outweighs the time spent evaluating the metric.

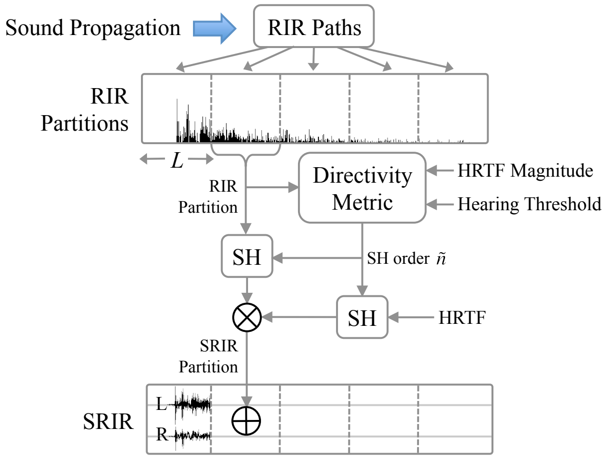


Figure 3: A visual representation of our spatial room impulse response construction algorithm. The RIR output of sound propagation is split into partitions with size L, and the directivity of each partition is evaluated to determine the minimum spherical harmonic (SH) order $\tilde{n}$ for the partition's spatial sound. The partition's RIR is converted to the SH basis up to order $\tilde{n}$, and then convolved with a SH representation of the HRTF up to order $\tilde{n}$. The resulting filters for the left and right channels are added to the output SRIR at the partition's offset in the IR.

Given M sound paths that arrived during a partition, the metric first computes the distribution of sound energy incident at the listener's position for each of the simulation frequency bands. The result is $X_{lm,b}$, a set of normalized SH coefficients for each simulation frequency band b up to a maximum SH order n_{max}. $X_{lm,b}$ can be computed using a form of Monte Carlo integration:

$$X_{lm,b} = \frac{1}{\sum_{j=0}^{M} I_{j,b}} \sum_{j=0}^{M} Y_{lm}(\mathbf{x}_j) I_{j,b}. \qquad (7)$$

Here, the basis functions are evaluated for each path's direction and then weighted by the path's intensity at each frequency band.

Next, we use this energy distribution and the user's HRTF to determine a magnitude response of the SRIR partition at each frequency band. As a preprocessing step, the frequency-domain HRTF $H(\mathbf{x}, f)$ is transformed into the SH basis to generate coefficients $h_{lm}(f)$. Then, the average magnitude response over each simulation frequency band b is computed to yield a spherical harmonic representation of the HRTF's magnitude response for that band, $h_{lm,b}$. Using the orthogonality property of the spherical harmonics (4), an approximation of the magnitude of the SRIR partition can be computed:

$$\left|\tilde{H}_{b,n}\right| = \sum_{l=0}^{n} \sum_{m=-l}^{l} X_{lm,b} h_{lm,b}, \qquad (8)$$

where $\left|\tilde{H}_{b,n}\right|$ is the pressure magnitude of the sound arriving during the partition for band b and SH order n. This relationship is used to efficiently evaluate the impact of using a given spherical harmonic order n on the resulting spatial sound. The goal of the metric is to determine SH order $\tilde{n} \leq n_{max}$ such that $\left|\tilde{H}_{b,\tilde{n}}\right|$ is perceptually indistinguishable from $\left|\tilde{H}_{b,n_{max}}\right|$. More precisely, the metric must satisfy the condition $\left|\left|\tilde{H}_{b,\tilde{n}}\right| - \left|\tilde{H}_{b,n_{max}}\right|\right| < \varepsilon$ where ε is a perceptually-based threshold.

One possibility is to compare against the absolute human threshold of hearing. The threshold is an important psychoacoustic quantity that corresponds to the smallest sound pressure level that a human can perceive at a given frequency [9, 23]. The threshold for the average adult listener, $T_q(f)$, can be analytically approximated as a function of frequency using the following relation [32]:

$$T_q(f) = 3.64(f/1000)^{-0.8} - 6.5e^{-0.6(f/1000-3.3)^2)} + 10^{-3}(f/1000)^4. \text{ (db SPL)} \tag{9}$$

We use this function to determine the maximum allowed error in the spatial sound for a given frequency band in units of pascals. Alternatively, the user's threshold of hearing can be measured using standard audiometric techniques and then interpolated to get the threshold at an arbitrary frequency. The final relationship that must be satisfied is then:

$$|p_b| \left| |\tilde{H}_{b,\tilde{n}}| - |\tilde{H}_{b,n_{max}}| \right| < T_q(b), \tag{10}$$

where $|p_b| = \sqrt{z_0 \sum_{j=0}^{M} I_{j,b}}$ is the total pressure magnitude for the partition. To determine the value of $\tilde{n}$ using the threshold of hearing, the metric starts at SH order $\tilde{n} = 1$, and then evaluates equation 10 for successively higher orders until the threshold is satisfied. The result is SH order $\tilde{n}$ that can be used to compute the SRIR for the current partition. If $\tilde{n} < n_{max}$, then significant computation can be saved.

4.2 Convolution with the HRTF

Once the minimum spherical harmonic order $\tilde{n}$ has been determined for a given partition, the next step is to convolve the partition RIR with the user's HRTF to generate the SRIR for the partition.

First, the time-domain spherical harmonic signal for the RIR partition must be computed from the M sound paths that arrived during the partition. This can be done by evaluating equation 7 for each time sample in the partition with the appropriate path delays added:

$$X_{lm,b}(t) = \frac{1}{\sum_{j=0}^{M} \delta(t-t_j) I_{j,b}} \sum_{j=0}^{M} \delta(t-t_j) Y_{lm}(\mathbf{x}_j) I_{j,b}. \tag{11}$$

The result of this operation is a set of normalized SH coefficients for each time sample and frequency band in the partition that represent an approximation of the directional information up to SH order $\tilde{n}$.

Next, the energy-time curve for the partition, $E_b(t)$, is computed as a sum of delayed impulses:

$$E_b(t) = \sum_{j=0}^{M} \delta(t-t_j) I_{j,b}. \tag{12}$$

This signal represents the sound energy decay for the partition at frequency band b.

To efficiently perform the convolution with the HRTF in frequency domain, the signals $X_{lm,b}(t)$ and $E_b(t)$ which are of length L must be padded at the end with zeros so that they are $2L$ audio samples long. In a preprocessing step, the HRTF is padded with zeros in time domain so that it is also $2L$ samples long. The HRTF is converted to frequency domain with a forward Fourier transform of size $2L$ and then projected into the spherical harmonic basis, yielding complex HRTF coefficients $h_{lm}(f)$. The partition SRIR for frequency band b can then be computed by convolving $h_{lm}(f)$ with the Fourier transform of the RIR signals:

$$p_b(t) = \mathscr{F}^{-1}\left[\sum_{l=0}^{\tilde{n}} \sum_{m=-l}^{l} h_{lm}(f) \mathscr{F}\left(X_{lm,b}(t)\sqrt{E_b(t) z_0}\right)\right] \tag{13}$$

where $\mathscr{F}$ is the Fourier transform operator. The resulting filters for all frequency bands are then band-pass filtered and then summed according to equation 6 to generate the full SRIR for the partition. Then, the partition SRIR is added to the output SRIR at the partition's time offset. When all partitions have been processed, the SRIR is complete and can be convolved with the anechoic audio for the sound source.

5 Implementation

In this section we describe the various implementation details for our spatial sound rendering system. We implemented our approach as a plugin for the Unity$^{\text{TM}}$game engine. The sound sources, listener, and scene geometry are specified by attaching scripts to objects within the game engine. Our system supports dynamic sources, listeners, and moving static geometry.

Sound Propagation: The propagation of sound within the virtual scene is computed in 4 logarithmically-distributed frequency bands: $0-110$Hz, $110-630$Hz, $630-3500$Hz, and $3500-22050$Hz. We use a hybrid of the image source method [5] for early specular reflections and Monte Carlo path tracing from the listener [28] for diffuse reflections. Potential specular reflection paths are found by tracing randomly distributed rays from the listener and specularly reflecting them up to a maximum order (e.g. 3 bounces) to sample possible reflection sequences [34]. This avoids an exponential explosion in the number of image sources. Then, the standard image source method is applied to the reflection sequences to validate the paths. Our sound propagation module also computes diffraction up to order 3 according to the UTD model [33, 29] during the specular ray tracing step. In the Monte Carlo path tracing, a different set of random rays is emitted from the listener and then reflected through the scene up to a high reflection order (e.g. 100 bounces) according to the BRDF(s) of the acoustic materials in the scene. The ray tracing for a single ray is terminated according to the adaptive threshold of [27], and we use temporal coherence techniques [29, 27] to improve the quality of the Monte Carlo path tracing by filtering the results for a path over several frames. The number of primary rays traced on each frame from the listener is calculated based on the time taken to compute the previous frame. This allows our system to adaptively reduce or increase the simulation quality to maintain a specific update time for sound propagation. About $500-1,000$ primary rays are traced for indoor scenes, while more rays are traced outdoors because most rays escape the scene after a few bounces. We used a target update time of 30 ms for all benchmarks. The ray tracing is parallelized across half of the available CPU threads (6 in this case), and these threads execute with low priority to avoid audio rendering glitches.

SRIR Construction: Efficient computation of our partition directivity metric requires fast evaluation of the real spherical harmonics. We use the formulation proposed in [30] that uses aggressive constant propagation and recurrence relations to speed up the computation for normalized cartesian vectors. In a recent study of HRTF localization, the 4th-order spherical harmonics were sufficient to achieve accurate localization performance [24]. As a result, we use a maximum spherical harmonic order of $n_{max} = 4$. In our implementation, we use a partition size of $L = 512$ samples, or roughly 10.7ms at a 48000kHz sampling rate. The filtering of the SRIR into frequency bands is accomplished using a 4th-order time-domain Linkwitz-Riley crossover network. The SRIR(s) for all sources are computed in parallel using the other half of the CPU threads (6 in our system). SRIR construction is performed in parallel with sound propagation in order to reduce the update period, and the RIR(s) are double-buffered such that the sound propagation for frame n is computed while the SRIR for frame $n-1$ is constructed.

Sharp Directivities: Low-order spherical harmonics may not always be sufficient to represent cases where the impulse response has very sharp directivities, such as with direct and early reflected sound. To handle this, we implemented a simple approach that finds important propagation paths in a preliminary pass over the RIR and then performs accurate HRTF interpolation (5) for just those paths. The other paths are computed using the approach from Section 4. A

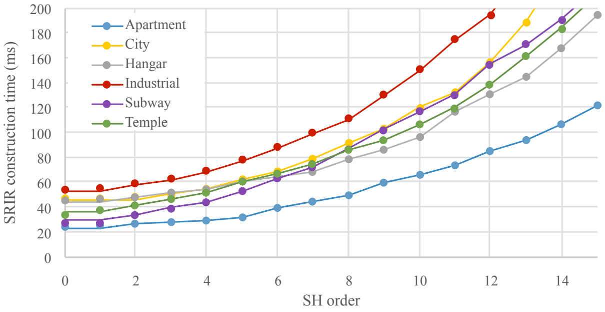


Figure 4: The performance of our spatial room impulse response construction algorithm scales quadratically with respect to the maximum spherical harmonic order n_{max}. We used $n_{max} = 4$ to generate the main results of our approach. The differences between the scenes are due to variation in the lengths of the impulse responses and the number of sources.

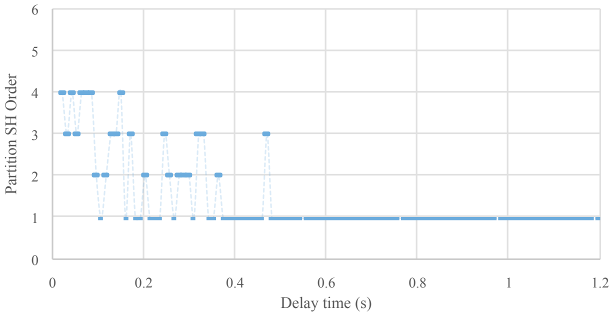


Figure 5: The spherical harmonic order $\tilde{n}$ determined by our directivity metric for each partition varies across the impulse response length. We observe a tendency for higher SH order toward the beginning of the IR where the pressure is greater and where there are more distinct paths with strong directivity (e.g. direct sound and early reflections).

path is considered important if its intensity is a significant fraction of the total energy in the impulse response, e.g. 1%. This approach tends to reduce the SH order required to represent strongly directional impulse responses. As a result, a smaller value for n_{max} can be used to save time in evaluation of the directivity metric. However, we did not notice any significant impact on performance or sound quality in the benchmark scenes so this module was disabled for our main results.

Auralization: Given a set of spatial room impulse responses, the auralization module uses a non-uniform partitioned block convolution algorithm to efficiently convolve the SRIRs with the audio for each sound source with low latency [11]. We use an initial block size of 64 samples, and then double the block size every 4 blocks until a maximum block size of 512 samples is reached. This keeps both the convolution latency (128 samples) and the latency to update the impulse response ($\leq$ 512 samples) low. A thread pool with 2 high-priority threads is used to execute the convolution for each group of 4 blocks in parallel, and the priority for each of the tasks is inversely proportional to the block size. On each audio rendering frame, the audio device output thread waits on the thread pool tasks that are due on that frame [3]. When the IR for a block is updated, a convolution is computed for both the previous and next filters, and then the results are interpolated in time domain over the block length [18]. The resulting audio for all sound sources is mixed and then sent to the audio device for playback. We use the Unity$^{\text{TM}}$game engine audio system which introduces an additional 21.3ms of latency due to audio output buffering.

6 Results and Discussion

The capabilities of our spatial sound propagation and rendering system were evaluated on 6 different scenes within the Unity$^{\text{TM}}$game engine. The scenes each have 6 to 9 sound sources, some of which are dynamic, and have geometry complexity typical of virtual reality and game environments. The scenes also contain interactive elements like moving doors that impact the resulting auralization.

The main results of our system on these scenes are summarized in Table 1. The times were measured on a 6-core 3.50GHz Intel i7-5930K machine by measuring the average time over the demo sequences in the supplementary video.

Performance:

The overall performance of our algorithm is reported for each scene in Table 1. By design, the time taken for sound propagation is about the same for all scenes, roughly 30ms. For the SRIR construction, our approach takes anywhere from 28.4ms to 68.8ms, whereas the previous approach takes 242.6ms to 488.0ms. This significant variation is mostly due to differences in the impulse response length and the number of sources in each scene. Differences in the directivity present in the scenes may also account for some of the variation. When the performance of the per-path approach is compared to ours, we observe a significant $6.7-9.1$ times speedup. This speedup enables our approach to update the SRIR fast enough for interactive applications. Our technique is able to satisfy perceptual latency thresholds of around 100ms, whereas the per-path approach is so slow that it introduces noticeable delay.

A significant parameter in the execution time of our algorithm is the spherical harmonic (SH) order at which the directivity is evaluated for each partition, n_{max}. In Figure 4 we show how the performance scales with respect to n_{max} for the six scenes. All scenes show a quadratic increase in execution time with respect to the SH order. This characteristic is mostly due to the evaluation of our directivity metric for every partition, not the cost of convolution with the HRTF. This is because our method tends to only use $\tilde{n}$ close to n_{max} for the first several partitions (see Figure 5). As a result, the increase in computational cost for the convolution of the RIR with the HRTF is smaller than the increase due to evaluation of the metric.

The primary benefit of our approach is that it enables the spherical harmonic order of the spatial sound to vary according to the directivity present in the room impulse response. This is illustrated in Figure 5. Toward the beginning of the impulse response where the directivity is stronger due to the presence of direct sound and early reflections, our approach tends to use a higher SH order. The SH order that is required decreases quickly thereafter due to the increasingly diffuse sound field. For the last half of the IR, 1$^{\text{st}}$ order is all that is needed to satisfy our directivity metric. This results in a large overall savings in computation versus using a fixed order for the whole impulse response. Our perceptually-based metric keeps the sound quality about the same as doing per-path HRTF interpolation, but has a much better overall performance.

Latency: There are many sources of latency in our system. We enumerate these in Table 2 and report an estimate of the total end-to-end latency of the audio pipeline based on the performance on the benchmark scenes. Sound propagation is responsible for roughly 30ms of latency, while SRIR construction can take $30-70$ms. There is an additional 10.7ms of latency for updating the convolution system with the new impulse response, while the convolution itself only adds 2.7ms of delay. Finally, a significant amount of la-

Table 1: The main results of our sound propagation and rendering system for the six benchmark scenes. We report the time taken for sound propagation separately from the SRIR construction time, and we compare the performance of our method to the performance of SRIR construction using per-path HRTF interpolation. Our method provides a speedup of $6.7-9.1$ over the previous approach.

	Scene Complexity		**Sound Propagation**	**Rendering**	**Per-path**		**Our technique**	
Scene	# Triangles	# Sources	Time (ms)	(% Real time)	SRIR time (ms)	SRIR time (ms)	Total (ms)	Speedup (Per-path/Ours)
Apartment	491,683	6	30.5	5.3	242.6	28.4	58.9	**8.6**
City	113,388	6	30.3	6.7	488.0	53.6	83.9	**9.1**
Hangar	473,328	7	31.4	7.1	449.9	53.8	85.2	**8.4**
Industrial	202,642	7	29.9	8.1	466.2	68.8	98.7	**6.8**
Subway	125,449	9	30.3	5.5	296.9	44.4	74.7	**6.7**
Temple	48,700	8	30.4	8.3	368.5	51.5	81.9	**7.1**

Table 2: The sources of latency in our sound propagation and rendering system. The total latency for our system is around the 100ms latency target needed for interactive virtual reality.

Scene	Latency (ms)
Sound propagation	29.9 - 31.4
SRIR construction	28.4 - 68.8
Convolution IR update	0 - 10.7
Convolution	2.7
Audio output buffer	21.3
Total	82.3 - 134.9

tency (21.3ms) is incurred by the lengthy audio device output buffer used by Unity^{TM}. The overall latency can range from 82.3ms to 134.9ms and the variation is strongly dependent on the scene. This is around the desired 100ms latency target for interactive audio. On the other hand, the per-path SRIR construction approach has a total latency of around 500ms for most of the scenes. This amount of latency is unacceptable for interactive applications and leads to noticeable delay and artifacts in the audio rendering with dynamic scenes.

It is important to note that the end-to-end latency of our system could be reduced further by using a shorter audio output buffer of just a few milliseconds (e.g. 64 samples, 1.3ms). The convolution IR update latency could also be reduced to 64 samples by using shorter FFT blocks for convolution with the source's audio, though this would decrease the performance of the convolution for long IRs.

7 User Evaluation

In order to evaluate the subjective impact of our spatial sound approach, we carried out a user evaluation in interactive virtual reality environments. The study compared the sound generated by our perceptually-based SRIR construction technique (Section 4), called *our* method, to the per-path HRTF interpolation approach, called the *base* method. For the *base* method, we tested two configurations: $base_s$, a single-threaded implementation (update time, $250-500$ms), and $base_p$, a parallel implementation where 10x as many threads are used to compute the SRIR (update time, $25-50$ms). Therefore, $base_p$ has about the same total latency as *our* approach, but uses 10x as much compute power, while $base_s$ has a latency about 10x that of both other methods.

The hypotheses of this study were: 1) $base_s$ has a much higher latency than *our* method and so there will be a preference for *our method*. 2) There will be no preference between $base_p$ and *our* method because the latency is similar and the sound is perceptually indistinguishable. 3) The strength of preference for either method will be dependent on the type of scene.

7.1 Study Design

The study was implemented using a within-subject experiment design and an A-B comparison protocol. Four different comparison conditions were evaluated: $base_s$ vs. *our*, *our* vs. $base_s$, $base_p$ vs. *our*, and *our* vs. $base_p$. These conditions were tested for 3 different scenes (City, Industrial, Temple), resulting in 12 different scenarios. Each scenario was repeated twice during the experiment, so each participant experienced a total of 24 trials. The trials were presented in a random order in two sets of 12 with a short break in-between. In each of these trials, the participant was presented an interactive audio-visual virtual reality experience where the sound was generated using either method A or method B according to the current condition under evaluation. During a trial, which lasted one minute, the participant was free to toggle between method A and B as many times as they wanted. The participant was spawned in the scene at a static position where they were able to freely move their head. The head movement was tracked using the head-mounted display and used to update the orientation of the listener in the virtual environment.

After each trial was completed, the participant answered a short subjective questionnaire to indicate their preferences on a 5-point Likert scale with respect to the following questions:

1. In which mode did the audio better correspond to the visuals?
2. In which mode could you better localize the sound?
3. Which mode was more realistic?
4. Which mode did you prefer?

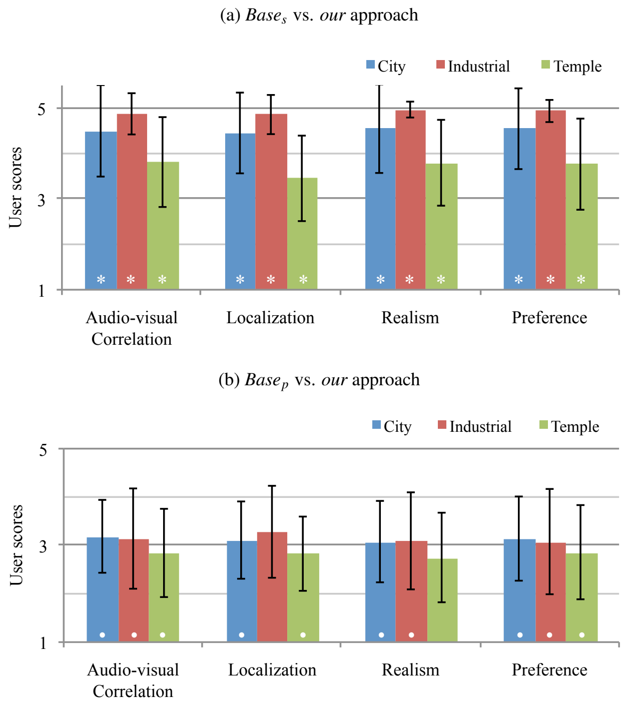
A response of 1 indicated a strong preference for method A, while a response of 5 indicated a strong preference for method B. A response of 3 means that the subject had no preferences.

7.2 System Details

The visual display of the virtual reality environment was presented using an Oculus Rift CV1^{TM}head-mounted display. The audio was delivered through the headphones that are integrated into the display. The study used a 14-core 2.60GHz Intel Xeon E5-2697v3 machine in order to render the audio for the $base_p$ case without glitches. The scenes were also simplified to contain just 1 or 2 sound sources so that the $base_p$ case would run in real time. A diffuse-field equalized version of the HRTF of subject 36 from the ARI HRTF database was used for all subjects [1].

7.3 Study Results

The questionnaire responses of the user evaluation are summarized in Figure 6. There was a total of 16 subjects who completed the study. The scores for the *our* vs. $base_s$ and *our* vs. $base_p$ conditions were reversed and combined with the scores for $base_s$ vs. *our* and $base_p$ vs. *our*, respectively. Subjects tended to answer all four questions with the same answer, so there is not much variation in responses among the questions.


(a) *Base$_s$* vs. *our* approach

(b) *Base$_p$* vs. *our* approach

Figure 6: The results of the user evaluation of our technique. We report the average response for each question and scene, where a value of 1 indicates a strong preference for the first method (*Base$_s$* or *Base$_p$*), a value of 5 indicates a strong preference for *our* method, and a value of 3 indicates no preference. The error bars correspond to the standard deviation. The * symbol indicates a significance with $p < 0.001$, while the • symbol indicates a lack of statistical significance ($p \geq 0.05$).

For the comparison between *base$_s$* and *our* method, the mean scores for all questions are between 4.45 and 4.97 for the City and Industrial scenes. These scenes contain fast-moving sound sources and so produce very noticeable delay or jumpiness when the sound is generated using the slow *base$_s$* method. For the Temple scene, the preference for our method is slightly less, with scores on all questions ranging from 3.45 to 3.80. This difference could be because the dynamic element in the Temple scene, an opening/closing door, moves slower than the sound sources in the other scenes. As a result, the latency differences between the methods are less noticeable in that scene. When analyzed with a one-sampled one-tailed Wilcoxon signed-rank test ($p < 0.001$) these results show a statistically significant preference for *our* method over *base$_s$* for all scenes and questions. This confirms our first study hypothesis that *our* method will be preferred over the *base$_s$* method. The differences between the Temple and Industrial/City scenes also supports our third study hypothesis that the preference will be dependent on the type of scene.

The other study comparison was between *our* method and *base$_p$*, the parallel version of the per-path SRIR construction technique. For this case, the mean scores for all scenes and questions are clustered between 2.75 and 3.28. A one-sampled two-tailed Wilcoxon signed-rank test ($p < 0.05$) was used to determine if there was any significant preference for either method. For all but two cases there is no preference between the methods ($p \geq 0.05$). For the Industrial scene, there is a small preference for *our* method on the localization question ($p = 0.026$), while for the Temple scene there is a small preference for the *base$_p$* method ($p = 0.035$) on the realism question. Overall, this confirms our second hypothesis that the differences between *base$_p$* and *our* method are not noticeable when the latency of the per-path SRIR construction is reduced by using 10x as many CPU threads.

8 Conclusions, Limitations, and Future Work

In this work we have presented a method for the efficient construction of the spatial room impulse response from the output of geometric sound propagation. The main contribution of our approach is a perceptually-driven metric based on the human threshold of hearing and the user's HRTF that adaptively determines the minimum required spherical harmonic (SH) order needed for different impulse response partitions. By using a lower SH order for partitions with more diffuse directivities, our algorithm saves considerable computation versus computing the SRIR for a fixed SH order or performing HRTF interpolation for each sound propagation path. We have evaluated the performance on several benchmark scenes and achieve a speedup of $6.7 - 9.1$ when compared to per-path SRIR construction. We have also performed a preliminary user study to compare the impact of our method versus per-path SRIR construction and see a significant scene-dependent preference for our method. As a result, our approach is able to compute spatial sound for the entire impulse response while meeting end-to-end system latency requirements for interactive virtual reality applications.

However, there are some limitations of our approach. Due to the use of 4th-order spherical harmonics, not all HRTF features may be represented accurately, and sharp directivities in the SRIR are not possible. This problem can be reduced somewhat by using a higher n_{max}, though this also results in more expensive evaluation of the partition directivity metric (see Figure 4). Another solution is to do full HRTF interpolation for important paths in the impulse response as described in Section 5. Since our directivity metric is applied to partitions, it is possible that the partitions may not be of sufficient resolution to capture variation in sound directivity at time scales less than one partition. This can be ameliorated by reducing the partition size, though this will result in a greater expense during convolution with the HRTF (Sections 4.2) because more smaller FFTs must be evaluated. Using a smaller partition size also has the drawback that more propagation paths are required to reduced the noise in the Monte Carlo directivity estimation (7). Finally, due to the use of geometric sound propagation for computation of the RIR, our system may not accurately model all acoustic effects like high-order diffraction and low frequency sound.

One avenue of future work is to perform a detailed evaluation to quantify the effect that sound propagation (as opposed to only direct sound) has on latency detection thresholds for interactive spatial sound.

Acknowledgements

The authors wish to thank the anonymous reviewers for their constructive feedback. We would also like to thank anonymous participants of the user study. This research was fully supported by Oculus & Facebook.

References

[1] Acoustics Research Institute of the Austrian Academy of Sciences. The ARI HRTF database. http://www.kfs.oeaw.ac.at/hrtf, accessed 2016-09-15.

[2] J. B. Allen and D. A. Berkley. Image method for efficiently simulating small-room acoustics. *The Journal of the Acoustical Society of America*, 65(4):943–950, April 1979.

[3] E. Battenberg and R. Avizienis. Implementing real-time partitioned convolution algorithms on conventional operating systems. In *Proceedings of the 14th International Conference on Digital Audio Effects. Paris, France*, 2011.

[4] J. Blauert. Spatial hearing: the psychoacoustics of human sound localization. *MIT Press*, 5:210–212, 1997.

[5] J. Borish. Extension to the image model to arbitrary polyhedra. *The Journal of the Acoustical Society of America*, 75(6):1827–1836, June 1984.

[6] A. Chandak, C. Lauterbach, M. Taylor, Z. Ren, and D. Manocha. Ad-frustum: Adaptive frustum tracing for interactive sound propagation. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1707–1722, 2008.

[7] C. L. Christensen and J. H. Rindel. A new scattering method that combines roughness and diffraction effects. In *Forum Acousticum, Budapest, Hungary*, 2005.

[8] J. J. Embrechts. Broad spectrum diffusion model for room acoustics ray-tracing algorithms. *The Journal of the Acoustical Society of America*, 107(4):2068–2081, 2000.

[9] H. Fletcher. Auditory patterns. *Reviews of modern physics*, 12(1):47, 1940.

[10] T. Funkhouser, I. Carlbom, G. Elko, G. Pingali, M. Sondhi, and J. West. A beam tracing approach to acoustic modeling for interactive virtual environments. In *Proc. of ACM SIGGRAPH*, pages 21–32, 1998.

[11] W. G. Gardner. Efficient convolution without input/output delay. In *Audio Engineering Society Convention 97*, pages 127–135. Audio Engineering Society, 1994.

[12] N. A. Gumerov and R. Duraiswami. A broadband fast multipole accelerated boundary element method for the three-dimensional helmholtz equation. *J. Acoustical Society of America*, 125(1):191–205, 2009.

[13] C. Hendrix and W. Barfield. The sense of presence within auditory virtual environments. *Presence: Teleoperators & Virtual Environments*, 5(3):290–301, 1996.

[14] K. H. Kuttruff. Auralization of impulse responses modeled on the basis of ray-tracing results. *Journal of the Audio Engineering Society*, 41(11):876–880, 1993.

[15] T. Lentz, D. Schröder, M. Vorländer, and I. Assenmacher. Virtual reality system with integrated sound field simulation and reproduction. *EURASIP Journal on Advances in Singal Processing*, 2007:187–187, January 2007.

[16] R. Mehra, L. Antani, S. Kim, and D. Manocha. Source and listener directivity for interactive wave-based sound propagation. *Visualization and Computer Graphics, IEEE Transactions on*, 20(4):495–503, 2014.

[17] R. Mehra, N. Raghuvanshi, L. Antani, A. Chandak, S. Curtis, and D. Manocha. Wave-based sound propagation in large open scenes using an equivalent source formulation. *ACM Trans. on Graphics*, 32(2):19:1–19:13, 2013.

[18] C. Müller-Tomfelde. Time varying filter in non-uniform block convolution. In *Proc. of the COST G-6 Conference on Digital Audio Effects*, 2001.

[19] E.-M. Nosal, M. Hodgson, and I. Ashdown. Improved algorithms and methods for room sound-field prediction by acoustical radiosity in arbitrary polyhedral rooms. *The Journal of the Acoustical Society of America*, 116(2):970–980, 2004.

[20] V. Pulkki. Virtual sound source positioning using vector base amplitude panning. *Journal of the Audio Engineering Society*, 45(6):456–466, 1997.

[21] B. Rafaely and A. Avni. Interaural cross correlation in a sound field represented by spherical harmonics. *The Journal of the Acoustical Society of America*, 127(2):823–828, 2010.

[22] N. Raghuvanshi, R. Narain, and M. C. Lin. Efficient and accurate sound propagation using adaptive rectangular propagation. *IEEE Transactions on Visualization and Computer Graphics*, 2009.

[23] D. Robinson and R. Dadson. Threshold of hearing and equal-loudness relations for pure tones, and the loudness function. *The Journal of the Acoustical Society of America*, 29(12):1284–1288, 1957.

[24] G. Romigh, D. Brungart, R. Stern, and B. Simpson. Efficient real spherical harmonic representation of head-related transfer functions. *IEEE Journal of Selected Topics in Signal Processing*, 9(5), 2015.

[25] J. Sandvad. Dynamic aspects of auditory virtual environments. In *Audio Engineering Society Convention 100*. Audio Engineering Society, 1996.

[26] L. Savioja. Real-Time 3D Finite-Difference Time-Domain Simulation of Mid-Frequency Room Acoustics. In *13th International Conference on Digital Audio Effects (DAFx-10)*, Sept. 2010.

[27] C. Schissler and D. Manocha. Adaptive impulse response modeling for interactive sound propagation. In *Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 71–78. ACM, 2016.

[28] C. Schissler and D. Manocha. Interactive sound propagation and rendering for large multi-source scenes. *ACM Transactions on Graphics (TOG)*, 36(1), 2016.

[29] C. Schissler, R. Mehra, and D. Manocha. High-order diffraction and diffuse reflections for interactive sound propagation in large environments. *ACM Transactions on Graphics (TOG)*, 33(4):39, 2014.

[30] P.-P. Sloan. Efficient spherical harmonic evaluation. *Journal of Computer Graphics Techniques*, 2(2):84–90, 2013.

[31] U. P. Svensson, R. I. Fred, and J. Vanderkooy. An analytic secondary source model of edge diffraction impulse responses . *Acoustical Society of America Journal*, 106:2331–2344, Nov. 1999.

[32] E. Terhardt. Calculating virtual pitch. *Hearing research*, 1(2):155–182, 1979.

[33] N. Tsingos, T. Funkhouser, A. Ngan, and I. Carlbom. Modeling acoustics in virtual environments using the uniform theory of diffraction. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 545–552. ACM, 2001.

[34] M. Vorländer. Simulation of the transient and steady-state sound propagation in rooms using a new combined ray-tracing/image-source algorithm. *The Journal of the Acoustical Society of America*, 86(1):172–178, 1989.

[35] D. N. Zotkin, R. Duraiswami, N. Gumerov, et al. Regularized hrtf fitting using spherical harmonics. In *Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA'09. IEEE Workshop on*, pages 257–260. IEEE, 2009.