Ambient Sound Propagation

ZECHEN ZHANG, Cornell University
NIKUNJ RAGHUVANSHI, Microsoft Research
JOHN SNYDER, Microsoft Research
STEVE MARSCHNER, Cornell University

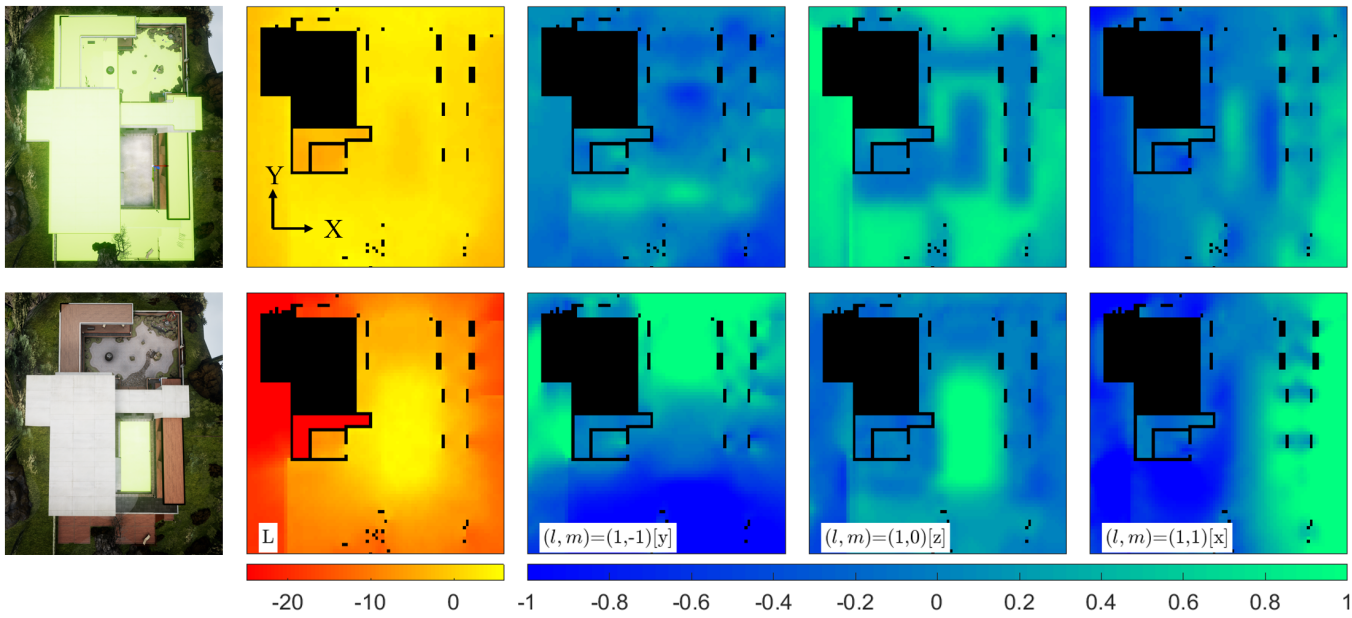

Fig. 1. Parameter fields in ZenGarden. We precompute the time-integrated directional energy arriving at every potential listener in a scene using real spherical harmonics (SH). Visual rendering is on left with extended sound source marked bright green; right panels show horizontal slices of our encoded 3D parameter fields. The figure's two rows show results for two ambient sources (marked in green) with separately encoded propagation effects: rain hitting the ground and rooftops (top) and rain hitting a rectangular water pool (bottom). Second column shows the total loudness field in dB. Remaining columns show first-order SH coefficient fields. We encode up to third order.

Ambient sounds arise from a massive superposition of chaotic events distributed over a large area or volume, such as waves breaking on a beach or rain hitting the ground. The directionality and loudness of these sounds as they propagate in complex 3D scenes vary with listener location, providing cues that distinguish indoors from outdoors and reveal portals and occluders. We show that ambient sources can be approximated using an ideal notion of spatio-temporal incoherence and develop a lightweight technique to capture their global propagation effects. Our approach precomputes a single FDTD simulation using a sustained source signal whose phase is randomized over frequency and source extent. It then extracts a spherical harmonic encoding of the resulting steady-state distribution of power over direction and position in the scene using an efficient flux density formulation. The resulting parameter fields are smooth and compressible, requiring only a few MB of memory per extended source. We also present a fast binaural rendering technique that exploits phase incoherence to reduce filtering cost.

CCS Concepts: • **Applied computing** → **Sound and music computing**; • **Computing methodologies** → **Virtual reality**;

Additional Key Words and Phrases: Diffraction, finite difference time domain (FDTD) wave simulation, flux density, head related transfer function (HRTF), incoherent extended source, interference, perceptual coding, spatialization, spherical harmonics (SH).

ACM Reference Format:
Zechen Zhang, Nikunj Raghuvanshi, John Snyder, and Steve Marschner. 2018. Ambient Sound Propagation. *ACM Trans. Graph.* 37, 6, Article 184 (November 2018), 10 pages. https://doi.org/10.1145/3272127.3275100

Authors' addresses: Zechen Zhang, zz335@cornell.edu, Cornell University; Nikunj Raghuvanshi, nikunjr@microsoft.com, Microsoft Research; John Snyder, johnsny@microsoft.com, Microsoft Research; Steve Marschner, srm@cs.cornell.edu, Cornell University.

0730-0301/2018/11-ART184 $15.00
https://doi.org/10.1145/3272127.3275100

1 INTRODUCTION

Ambient sounds like wind, rain, or surf provide a dynamic background, propagating through a 3D scene to complement visuals and immerse the listener in an environment. As the listener navigates, loudness changes convey the size and shape of the sound source. For instance sound attenuates when moving away from the beach but not when moving along it. Variation due to scene occlusion indicates how open or enclosed the surrounding space is. Directionality from sound streaming through portals or adjoining chambers reveals their presence even when behind the listener. The beach sounds big outside but becomes directionally crisp when heard through an open door or window. We capture and render these effects for the first time, within a small runtime budget appropriate for background sounds in games and virtual reality (VR).

We assume ambient sources superpose many independent, point-like elementary sound events, such as the impact of each drop in a rain shower. Modeling these events individually is impractical in interactive applications which typically support about a hundred active sounds in total and allocate only a few for the background. Replacing this complexity by a few point proxies causes unrealistic wobbles in loudness as the listener moves past the proxies and fails to reproduce the correct aggregate effects of distance falloff and shadow softening.

When the listener is unable to distinguish these individual sound events, we observe that ambient sources can be idealized as incoherent in space and time. This idealization fails for individual conversations heard at a crowded party or for nearby cars on a street corner, but it suffices as the crowd's chatter or highway noises merge at greater distance. Prior work on propagation from extended sources has instead focused on spatially coherent sources at most a few meters across, such as free-field radiation from vibrating shells [James et al. 2006] and scattering/shadowing from discrete objects in sparse outdoor scenes [Mehra et al. 2013]. Coherent wave fields are highly oscillatory over space, making these techniques compute- and memory-intensive. The (time-averaged) power distribution of an incoherent field is smooth, saving CPU and RAM.

Given the source's spatial power distribution, we precompute a single 3D wave simulation using the finite-difference time-domain (FDTD) method. This Eulerian approach naturally accounts for diffraction and scattering, and it propagates the entire field at once, making precomputation time insensitive to the source's spatial extent. Our first contribution is the notion of an incoherent wave source and an efficient formulation that generates phase-decorrelated, bandlimited noise signals, yielding response fields with smooth power variation, avoiding spatial oscillations from interference.

Simulation output is a pressure field over time and space; it captures how sound propagates from the source but is too large to store in raw form and does not make the information needed at runtime readily available. Our second contribution is a streaming encoder that efficiently extracts and stores compact perceptual information for incoherent fields. We compute a 3D grid of low-order spherical harmonic (SH) coefficients representing the directional distribution of acoustic power at points regularly sampled throughout the scene. Our encoder computes the flux density vector at each timestep and sample point, and incrementally accumulates the SH coefficients. Prior techniques [Laitinen et al. 2012; Raghuvanshi and Snyder 2018] also apply flux but on short, coherent signals for which overlapping arrivals in different directions cannot be discriminated. With a sustained, incoherent source signal, we show for the first time that flux can tease apart the steady-state spherical power distribution.

Third, we show that incoherence can be exploited for faster binaural rendering at runtime. Existing techniques require expensive convolution with the HRTF (head related transfer function) to preserve interaural phase differences. But in an ideally incoherent sound field, phase relationships are difficult to sense and frequency-dependent interaural loudness differences dominate. These can be achieved cheaply using standard parametric equalization. We obtain good results with just four frequency bands, by equalizing the mono-aural representative sound to be propagated using 4-channel left/right gains computed for the current listener position and head orientation in the scene.

Overall, ours is the first system to model salient directional propagation effects from arbitrarily large ambient sources in complex game environments. Precomputation takes about several hours on a single desktop. Runtime cost per extended source is less than 1MB of memory and about 100μs per-frame computation on a single core. Cost is largely insensitive to source size or scene's geometric complexity. Our technique integrates transparently with standard game engines like Unreal Engine 4.

2 PRIOR WORK

Computer graphics. Light transport methods have long supported extended sources to provide realistic illumination and shadows. Approaches based on ray sampling [Cook et al. 1984; Hasenfratz et al. 2003; Ritschel et al. [n. d.]] require many hundreds of rays per pixel in complex scenes with large sources. Precomputed radiance transfer (PRT) [Sloan et al. 2002] allows real-time manipulation of distant environmental lighting. Its use of scene precomputation is notably similar to ours; just as PRT reduces spatial sampling requirements by assuming smooth lighting and representing it with low-order SH, so we apply SH to the coarse directional distribution of power at an arbitrary listener in the scene.

These approaches ignore diffraction, whose effects are much more perceptible at audible than visible wavelengths (e.g., visually occluded sounds often remain audible). By performing wave simulation, our approach accounts for diffraction and wave scattering from complex geometry. Furthermore, its cost is determined by the temporal and spatial discretization (see Section 4) and is nearly independent of the complexity of scene geometry or source size.

Wave-based rendering. [Moravec 1981] proposed a visual rendering framework based on wavefront tracking. It sweeps a single, monochromatic (and thus, coherent) plane wavefront through the entire scene. Propagation becomes spatial convolution that yields complex field coefficients on a target plane wavefront, which can jump any distance analytically in free space but must be stepped incrementally in the presence of geometry. Diffracted occlusion is implicitly modeled (via the Kirchoff approximation valid at high frequencies) but scattering off geometry is ignored. Complex source distributions would require numerous sweeps from different incoming directions. The paper speculates that interference artifacts in the

results could be ameliorated by averaging over multiple simulations with randomized phase.

Geometric acoustics. Geometric acoustic systems rely on ray tracing for sound by invoking a high frequency approximation to the wave equation, an approach studied extensively in room acoustics and computer graphics. Modeling diffraction exacerbates path sampling requirements compared to CG light tranport [Cao et al. 2016], and a general system that handles arbitrary-order diffraction and scattering in complex scenes remains beyond the state of the art; consult the review and discussion in [Savioja and Svensson 2015]. Recent work in CG precomputes edge visibility to accelerate the diffraction computation for point sources [Schissler et al. 2014] and some online geometric systems support extended sources but can consume multiple machines' worth of computation for simple scenes of a few thousand polygons [Schröder 2011].

Wave solvers. Finite-Difference Time-Domain (FDTD) simulation is widely used in acoustics [Hamilton et al. 2017; Kowalczyk and Van Walstijn 2011] and electromagnetics [Taflove and Hagness 2005]. It is relatively easy to implement but subject to non-linear dispersion that accumulates phase errors as wavefronts propagate. For coherent signals, fine discretization is required to ensure phase accuracy. Since we ignore output phase, dispersion is not a major concern but numerical dissipation, which affects energy propagation, must still be controlled. Further details are explained in Section 4.

Coherent point sources. Precomputed wave approaches [Raghuvanshi and Snyder 2014, 2018; Raghuvanshi et al. 2010] trade runtime CPU for RAM needed to store the output field. High-order diffraction and scattering are modeled naturally. Geometry of any complexity is uniformly handled by voxelizing onto the simulation grid; aliasing is eliminated by ensuring source signals are appropriately bandlimited. Our general approach is similar but treats extended incoherent rather than coherent point sources.

Coherent extended sources. The equivalent source method approximates a known wave field using a linear combination of elementary multipole solutions to the wave equation. It has been used for modeling free-field radiation [James et al. 2006] and environmental propagation [Mehra et al. 2013] for coherent extended sound sources a few meters across, like vibrating shells. Coherent wave fields are highly oscillatory over space, requiring numerous multipoles to fit well and making such techniques heavyweight, especially at high frequencies. Recent developments focus on controlling the CPU cost of summing multipole contributions [Chadwick et al. 2009] and on reducing memory [Li et al. 2015]. For incoherent sources, chaotic phase can be ignored while time-averaged power varies smoothly, admitting a more compact approximation.

Sound texture synthesis. Sound textures arise from a collection of spectrally similar events which overlap in time. [McDermott et al. 2013; McDermott and Simoncelli 2011] study perceptually important statistics from real recordings, synthesizing new sound textures and assessing their recognizability. This work is applicable to generate better representative sounds (see Section 3) in our system. Our work instead focuses on propagation effects in virtual environments.

Binaural rendering. [Schissler et al. 2016] directionally renders large extended sources in the free field, ignoring the effect of scene geometry which is our focus. The technique accelerates in part by assuming arrivals from different directions are perfectly in phase. This is hard to arrange for real extended sources. We make the opposite and physically-motivated idealization that phases of arrivals in different directions are completely decorrelated, letting us ignore phase entirely and consider only the HRTF's magnitude spectrum.

Like our method, [Raghuvanshi and Snyder 2018] captures directional effects but for point sources. It encodes directionality separately for the direct sound (via loudness and a direction vector) and for early reflections (via six simple basis functions around the coordinate axes). We represent directional power averaged over all time, in terms of the simple but general SH basis. Both that method and ours perform directional analysis using flux which is known to merge directions when wavefronts arrive nearly simultaneously. For an incoherent source, one expects multiple wavefronts to arrive chaotically all the time, confusing flux. We demonstrate for the first time that it works well in our incoherent regime. We also reduce runtime rendering cost by avoiding costly convolutions required to render phase information.

3 SYSTEM OVERVIEW

The designer tags surfaces or volumes in a 3D scene as the extended source. Our goal is to play back in real time a (usually pre-recorded) *representative sound* as if it were emanating from that source and propagating through that scene. Loudness and directionality cues should change in a smooth and natural way as the listener moves.

Our system works in two stages. In the precomputation stage, we voxelize a 3D domain containing the extended source and scene geometry and introduce sustained, volumetric noise. This noise signal is entirely synthetic and used to predict sound transport effects, with no relationship to the representative sound played at run-time. A wave simulation then propagates this noise throughout the scene. Parameter fields are encoded representing the time-averaged power distribution as a (5D) function of position and direction at the listener, where directionality is represented using low-order (n=4) SH. These fields then modify the representative sound clip at playback time given the listener's changing position and orientation.

The emitted power distribution, shape, and location of the extended source is baked in with respect to the static scene and can't be changed at runtime. Any input clip can be used at runtime to represent the source sound; it should be suitably "ambient" with loudness and spectral content not changing too abruptly.

4 PRECOMPUTED SIMULATION

We use the finite difference time domain (FDTD) numerical method [Taflove and Hagness 2005] to simulate wave propagation. It models air pressure change over time by solving the wave equation

$$\frac{1}{c^2}\frac{\partial^2 p}{\partial t^2}(x,t) - \nabla^2 p(x,t) = s(x,t) \tag{1}$$

where x is 3D spatial location, t is time, c=340 m/s is the speed of sound, and p is acoustic pressure to be solved for. The (input) forcing term $s(x,t)$ represents the source perturbation to be propagated and will be detailed in Section 5.

FDTD is simple to implement. Its major disadvantages are numerical dispersion, causing different frequencies to incorrectly travel at different speeds, and numerical dissipation, causing high frequencies to be attenuated. As discussed later in Section 6, the acoustic parameters we're interested in depend on the response signal's time-averaged power and neglect its phase, making them insensitive to dispersion. Dissipation thus forms the main limit on the highest frequency we can simulate for a given grid spacing.

Simulation cost is proportional to the product of scene volume, spatial resolution, temporal resolution, and duration. Output quality depends on resolution, which determines the field's spatial detail, and the number of frequency bins resolvable in the band where the source propagates without artifacts, which determines the amount of randomness we can inject. We fix the spatial grid spacing (Δx) and number of frequency bins (ΔN) to achieve the required quality, and determine the other parameters as follows.

With specified grid spacing Δx, the time step is

$$\Delta t = \frac{C\,\Delta x}{c}, \tag{2}$$

where the Courant number $C = 1/\sqrt{3}$ is the maximum ensuring CFL stability, and the Nyquist frequency is

$$f_n = \frac{1}{2\,\Delta t}. \tag{3}$$

A simulation with duration T requires N_t time steps

$$N_t = \frac{T}{\Delta t}. \tag{4}$$

The simulated pressure response at any grid point thus yields a corresponding number of "bins" or frequency samples in its minimal discrete Fourier transform given by

$$N = \lfloor N_t/2 \rfloor + 1. \tag{5}$$

We use standard leap-frog integration, in which dissipation is greatest along axial directions [Kowalczyk and Van Walstijn 2011; Schneider and Wagner 1999]. Dissipation can thus be controlled by ensuring our source signal's content is no higher than the axial cutoff frequency [Schneider and Wagner 1999]. We denote this cutoff f_d ($f_d < f_n$), defined as

$$f_d = \frac{1}{\pi\,\Delta t}\sin^{-1}(C). \tag{6}$$

Low frequencies in the source signal should also be removed: a non-zero DC component generates a non-zero particle velocity (i.e. wind) in the direction away from the source. We therefore limit the power spectral density (PSD) content of the source signal to $[f_{\min}, f_{\max}] \subset (0, f_d]$ and denote its bandwidth $\Delta f = f_{\max} - f_{\min}$.

The number of frequency bins included in this band is then

$$\Delta N = N\,\frac{\Delta f}{f_n}. \tag{7}$$

We'll see in the next section that the bigger ΔN, the more the simulation approaches ideal incoherence. The duration of the source signal is $T_S = N/f_n = \Delta N/\Delta f$. Extra time steps are added after the emitted source signal ends to let it finish propagating through the scene. The padded time duration is $T_D = D/c$ where D denotes the length of the diagonal of the cuboid simulation domain, yielding the total simulated time $T = T_S + T_D$.

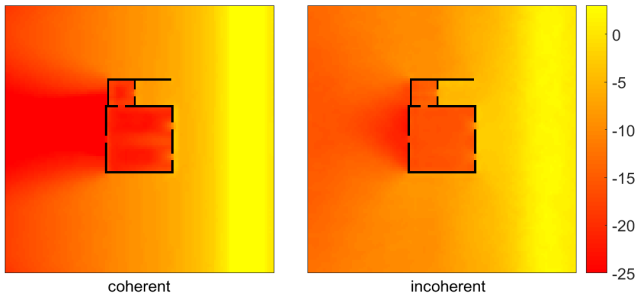

Fig. 2. Total loudness (dB) due to coherent vs. incoherent extended source.

We set $\Delta x = 0.255$m, yielding the sampling rate $1/\Delta t = 2309$Hz and dissipation cutoff $f_d = 452$Hz. We set the source PSD's non-zero band as $[f_{\min}, f_{\max}] = [62.5, 400]$Hz and the number of frequency bins included in it as $\Delta N = 1000$. This yields $N_t \approx 9200$ in our experiments. Our simulation is computed on a cuboid domain with perfectly matched layer (PML) [Rickard et al. 2003] absorber to suppress spurious reflections from the domain boundary.

5 INCOHERENT SIGNAL SYNTHESIS

A key component of our system is signal generation for an extended, spatio-temporally incoherent source, representing $s(x, t)$ in (1). Figure 2 shows the problems that arise from a coherent source. The scene in this experiment is a two-story beach house facing a much longer ocean surf source. When all points covered by the source emit a phase-locked Gaussian derivative pulse, an overly bright band of constructive interference forms near the source (right of image) and fringes (spatial oscillations) in the occluded area behind the house. Inside the house, note the increased spatial variation and overall attenuation compared to the incoherent case. Our new incoherent source yields a smoother and more physically motivated total loudness field.

Coherence is the tendency of wave field observations at different times and places to be correlated. We design our source to exhibit complete spatial incoherence and as little temporal coherence as possible while satisfying the frequency content constraints discussed in the previous section.

Spatial incoherence. A source much smaller than the sound wavelength can be treated as a point. When it is bigger, the signal it emits must be decorrelated over its extent as well as over time. To ensure spatial incoherence, we simply generate independent signals in each FDTD grid cell covered by the source. This approach is simple and achieves spatial incoherence up to the grid resolution; the smaller the voxels, the more closely the result approximates perfect spatial incoherence.

Temporal incoherence. A sequence of independent, zero-mean random samples is fully incoherent but isn't bandlimited and so incurs numerical dissipation. To reason about bandlimiting the signal appropriately for simulation, we consider the discrete Fourier domain, where this ideal white noise exhibits constant amplitude and independently random phase at each frequency bin. The signal we desire has the same random phase and constant amplitude but only across

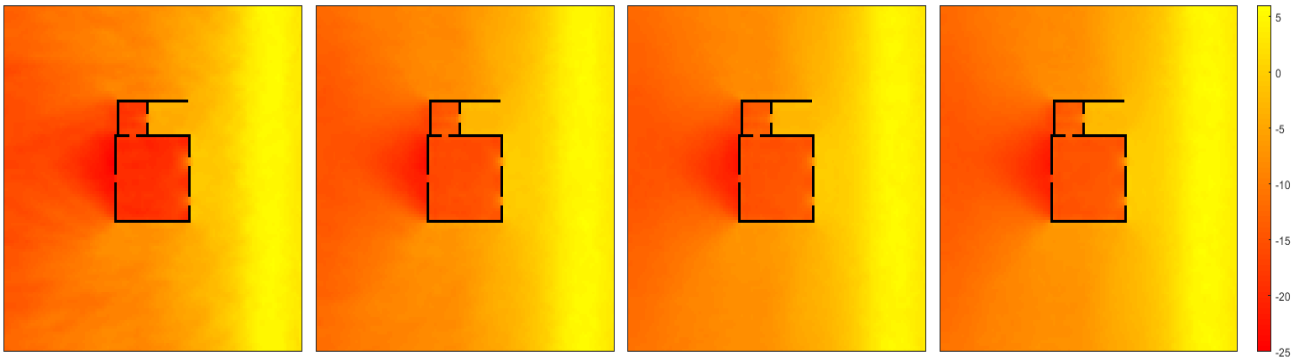

Fig. 3. Convergence with increasing ΔN. From left to right: $\Delta N = 63, 250, 1000, 4000$, visualizing total loudness in dB scale.

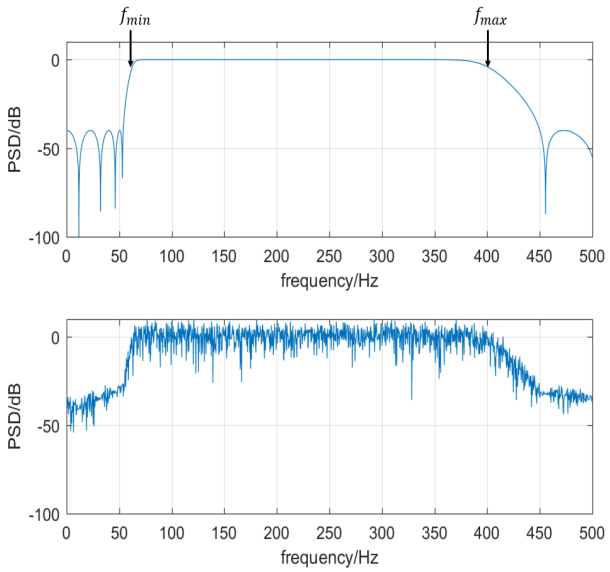

Fig. 4. Online incoherent synthesis. Top plot is the desired magnitude response of the source signal; bottom is its PSD when actually generated via on-the-fly IIR filtering of white noise.

the ΔN frequency bins in the available band $[f_{\min}, f_{\max}]$. A longer simulation increases ΔN and thus incoherence. Figure 3 shows how the simulation converges to a smooth result as ΔN increases.

Online synthesis. The most straightforward way to generate the source signal is via Fourier synthesis: for each voxel, construct the discrete signal in Fourier space by setting the magnitude of each frequency bin to the PSD target at that frequency, and its phase as a random number, then take the inverse Fourier transform. This method precisely controls the spectrum, but must store the full temporal signal at every source voxel. For large sources, this overwhelms storage for the FDTD state itself, which only updates pressure using the field at current and prior time step.

To reduce memory, we synthesize the source signal by filtering zero-mean white noise on-the-fly during simulation. The resulting source signal is $s(t, x) = w(t, x) * h(t)$ where $*$ denotes time-domain convolution. Note that the filter $h(t)$ is independent of position; we assume elementary sources share the same spectral content.

We implement h as an order-n_h IIR filter, which accesses samples from the most recent n_h timesteps. The memory required is proportional to the product of the number of source grid cells and the filter order n_h, independent of the simulation's duration. Figure 4 compares PSDs between the target and the actual stochastic signal.

Filter details. For a similar rolloff factor in the transition between pass-band and stop-band, an IIR filter requires lower order compared to an FIR filter. It incurs more phase distortion, but this is not troublesome because the signal's phase is randomized.

We apply a band-pass Chebyshev type-II filter [Smith 2008]. The pass band is set as $[62.5, 0.36\,C\,f_n]$ in Hertz, and the stop bands as $[0, 20]$ and $[0.675\,C\,f_n, f_n]$. This guarantees the pass-band is contained in the non-dissipative frequency range. The ripple (ratio between the largest and smallest magnitudes of filter's frequency response) allowed in the pass band is 6dB and the attenuation factor of the stop bands is 40dB. We use filter order $n_h = 16$. Memory demanded by online source synthesis is below one megabyte even for an extended source comprising tens of thousands of voxels.

6 ENCODER

At each simulation voxel, our encoder computes flux at the next simulation time step and accumulates its low-order SH power distribution. The resulting parameters fields are then spatially downsampled and compressed as in [Raghuvanshi and Snyder 2014].

Acoustic flux density. Flux density (or simply, flux) represents instantaneous power transport in the fluid over a differential oriented area, analogous to irradiance in optics. It estimates the direction of a wavefront passing x at time t, via

$$f(x,t) = p(x,t)\,v(x,t), \quad v(x,t) = -\frac{1}{\rho_0}\int_{-\infty}^{t} \nabla p(x,\tau)\,d\tau \qquad (8)$$

where v is the particle velocity and ρ_0 is the mean air density (1.225kg/m^3). We use central differences to compute spatial derivatives for ∇p, and the midpoint rule for numerical time integration, as in [Raghuvanshi and Snyder 2018].

We recover the time-averaged directional power distribution at any spatial point x (suppressed below) as follows. At every time

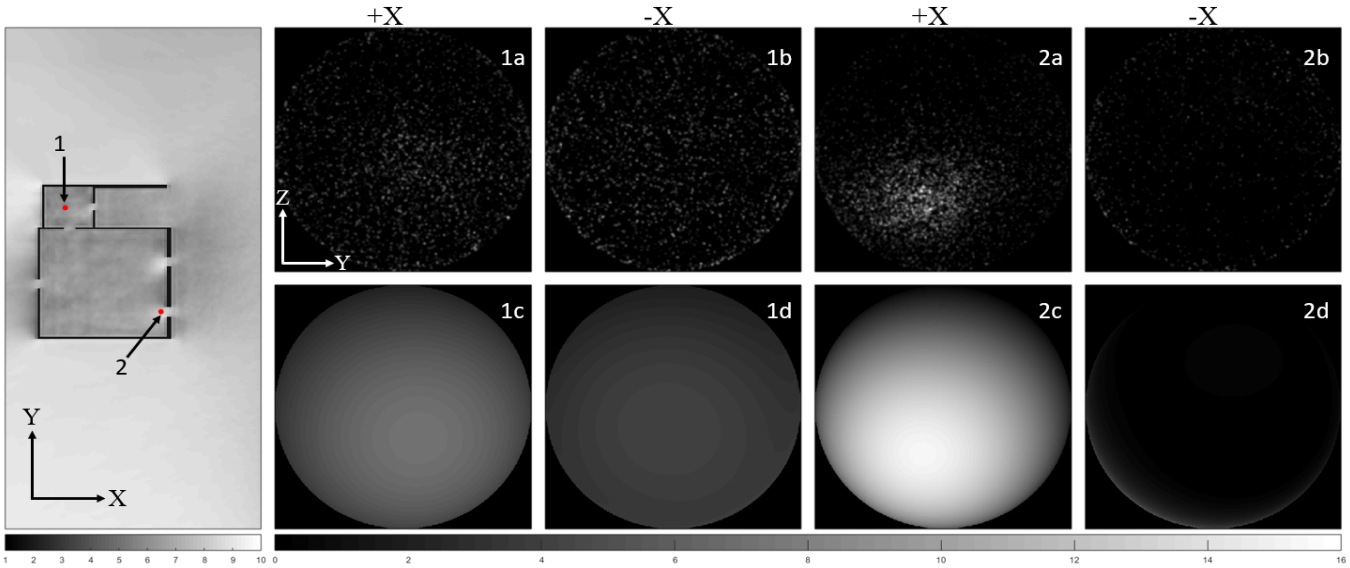

Fig. 5. Directional analysis example. Left column: overall directionality. Right four columns: recorded directional RMS power distribution $\sqrt{E(\Theta)}$ for the two marked locations. (1a)/(1b) show the $\pm X$ hemispheres of the directional power distribution at location 1. (1c)/(1d) show the corresponding SH reconstruction. (We visualize $\sqrt{E(\Theta)}$ instead of $E(\Theta)$ to emphasize low-power spots.) (2a)-(2d) show the same analysis for location 2.

step t, take the instantaneous flux to form the unit vector $\hat{f}(t) \equiv f(t)/\|f(t)\|$ and associate the instantaneous power $p^2(t)$ to that single direction, followed by time-averaging, yielding

$$E(\Theta) = \frac{1}{T}\int_0^T p^2(t)\,\delta(\Theta - \hat{f}(t))\,dt \tag{9}$$

where Θ represents a direction and $\delta(\Theta)$ is the Dirac delta function in direction space.

SH projection. We encode directional power distribution as a smooth spherical function by projecting to order-n SH via

$$E_{l,m} = \frac{1}{T}\int_0^T p^2(t)\,Y_{l,m}(\hat{f}(t))\,dt. \tag{10}$$

where $Y_{l,m}$ are the n^2 real SH basis functions [Sloan 2013]. We use $n = 4$. We note that the integral in (10) can be evaluated progressively without storing history for $p(t)$ and $f(t)$. Our streaming encoder accumulates into the n^2 SH coefficients $\{E_{l,m}\}$ at each time step. A smooth reconstruction of input power distribution is then given by

$$E(\Theta) \approx \sum_{l=0}^{n-1}\sum_{m=-l}^{m=l} E_{l,m}\,Y_{l,m}(\Theta). \tag{11}$$

Windowing. To avoid directional ringing [Sloan et al. 2002], we filter the $\{E_{l,m}\}$ through the Kaiser window [Oppenheim 1999]

$$K(l) = \frac{I_0\left(\beta\sqrt{1-(l/n)^2}\right)}{I_0(\beta)}. \tag{12}$$

This is an efficient approximation of the theoretical window (DPSS) maximizing main lobe energy. Here, I_0 is the zeroth-order modified Bessel function of the first kind and β is a positive parameter adjusting the shape of the window. We set $\beta = 5$.

The final SH coefficient becomes

$$E'_{l,m} = K(l)\,E_{l,m}. \tag{13}$$

We assume windowing and drop the prime in the following.

Normalization. The DC component, $E_{0,0}$, corresponds to total power received at the listener. A scale factor is applied to all coefficients $\{E_{l,m}\}$, such that the maximum loudness (see below) over the iso-surface 1m away from the source is normalized to 0dB.

Example. Figure 5 visualizes RMS power distribution $\sqrt{E(\Theta)}$ and its SH reconstruction at two listener locations in BeachHouse. A simple metric for overall directionality used in the figure's left column is the ratio between the L2 norm of the linear SH coefficients and the DC component. The figure also shows the power distribution for two listener positions. Location 1 lies inside the small reverberant room and is more directionally diffuse compared with location 2 near the portal facing the extended source. Our compact SH representation is able to capture such strong anisotropy introduced by portals and corners.

Compression. As with other energetic acoustic parameters [Raghuvanshi and Snyder 2014], our encoded SH coefficients $\{E_{l,m}(x)\}$ form a spatially smooth field. We apply a similar pipeline of spatial smoothing, quantization and compression.

We encode the DC component in logarithmic space via total loudness $L = 10\,\log_{10} E_{0,0}$, and the higher-order SH coefficients $l > 0$ in linear space relative to DC via $E_{l,m}/E_{0,0}$. Representing total loudness in decibels accords with human perception; encoding relative

SH coefficients retains directional information even in highly occluded cases. L is clamped within the range $[-60,\ 6]$dB. The rest of the coefficients are bounded by the ratio of the max value of the SH basis function to DC (i.e. by $|Y_{l,m}(0,0)/Y_{0,0}|$) for non-negative spherical functions; $[-2,\ 2]$ encompasses the range of values we've encountered for windowed functions with $n = 4$.

The parameters are then spatially down-sampled using simple averaging over a 1m×1m×1m cube centered at x. Only parameters in voxels unoccupied by scene geometry and visible to x are included. Parameters are quantized using the quantum 1dB for L, which is the just-noticeable-difference for loudness [ISO3382 2009], and 0.04 for the others. Finally, the parameter fields are compressed along each x scanline (as with PNG images) and (lossless) LZW applied to the running difference. This pipeline attains a compression factor of over a million with respect to the raw parameter fields, yielding an output of about a megabyte per ambient source in a scene. The wave simulation data, whose size is on the order of terabytes, does not need to be stored by our streaming encoder but is processed at each time-step and discarded.

7 RUNTIME

Runtime decoding is similar to [Raghuvanshi and Snyder 2014]. Each of the n^2 decoded parameters is trilinearly interpolated over the visible voxels of the surrounding cube's 8 vertices to the current listener position, yielding the propagated directional power distribution $E(\Theta)$.

Spatialization. The human auditory system relies on interaural phase (IPD) and loudness difference (ILD) cues for localizing sounds in direction space. The HRTF (head-related transfer function) captures the mutual phase shift and shadowing introduced by the human head and shoulders at the two ears, for incoming coherent wavefronts over various directions. For each direction, it tabulates the complex transfer function for both ears at a dense (~200) set of frequency bins. Binaural rendering is performed by taking a source's mono-aural emitted signal and convolving it with the HRTF via complex multiplies at each frequency bin. This is computationally costly, but necessary for point sources as they radiate spatially coherent wavefronts with a salient IPD.

For chaotic and extended sources, we observe that phases of the arriving field in different directions tend to be mutually uncorrelated, making IPD cues less detectable. We thus render only the frequency-dependent head shadowing effect (ILD) given the runtime listener head pose and the incoming spherical power distribution, $E(\Theta)$.

In the frequency domain, denote the (complex-valued) HRTF as $H(\Theta, f)$ where Θ is sound arrival direction and f is frequency. We set the gain of the sound signal at frequency f as

$$g(f) = \sqrt{\int_{\Omega} E\left(\mathcal{R}^{-1}(\Theta)\right) \left\|H(\Theta, f)\right\|^2 \, d\Theta}, \tag{14}$$

where Ω is the direction space, E is the reconstructed directional power distribution from (11), and $\mathcal{R}$ transforms directions from the head to the world coordinate system.

We divide the audible frequency range into n_H sub-bands. For the i-th band $[f_0^i,\ f_1^i]$, the respective gain is

$$g^i = \sqrt{\int_{\Omega} E\left(\mathcal{R}^{-1}(\Theta)\right)\ H^i(\Theta)\, d\Theta}, \tag{15}$$

where $H^i(\Theta)$ is the average HRTF power over the sub-band

$$H^i(\Theta) \ = \ \frac{1}{f_1^i - f_0^i} \int_{f_0^i}^{f_1^i} \left\|H(\Theta, f)\right\|^2 \, df. \tag{16}$$

Representing H^i using the same low-order SH approximation used for the propagated power distribution E, the spherical integral in (15) becomes a simple dot product of a pair of length n^2 vectors, followed by a square root. Finally, the resulting n_H scalar gains are applied to the representative clip, which is separated into n_H sub-bands using online equalization filters. Our implementation uses the built-in equalizer from the XAPOFX library.

We use $n_H = 4$ sub-bands having center frequencies f_c at 125, 600, 2400, and 9600Hz, with two-octave bandwidth, $[f_c/2, 2\, f_c]$. The SH projection of the HRTF is done as a precomputation (see below) and does not change at runtime.

SH rotation. We currently support azimuthal rotation of the listener; general rotation is a simple extension [Kautz et al. 2002]. If the listener's head is at azimuthal angle θ, the block-diagonal SH rotation matrix is

$$M(\theta) = \text{diag}\{M_0(\theta);\ M_1(\theta);\ \ldots;\ M_{n-1}(\theta)\}, \tag{17}$$

where $M_k(\theta)$ is the $(2k+1) \times (2k+1)$ matrix

$$(M_k)_{ij}(\theta) = \begin{cases} \cos((k+1-i)\,\theta), & \text{if } i = j, \\ \sin((i-k-1)\,\theta), & \text{if } i + j = 2k+2, \\ 0, & \text{otherwise,} \end{cases} \tag{18}$$

for $i, j = 1, 2, \ldots, 2k+1$. This matrix transforms the SH vector E from the world to the head coordinate frame, where it is dotted with each of the n_H HRTF vectors to yield the gains.

HRTF projection. We use the public domain CIPIC database [Algazi et al. 2001]. After converting HRIR measurements to frequency domain HRTFs $H(\Theta,\ f)$ via the discrete Fourier transform, we project the average HRTF power in each sub-band to SH via least-squares optimal projection [Sloan et al. 2003]. Note that measurements in the database have sampling gaps around the poles.

Figure 6 shows our representation based on average HRTF power distribution $H^i(\Theta)$ and its corresponding SH reconstruction.

8 RESULTS

Precomputed simulation and streaming encoding are performed on a single desktop with Intel i7 CPU @ 3.70GHz and 32G RAM. The technique is integrated with Unreal Engine 4™. Precomputation data for our scenes is summarized in Table 1. Bake times vary proportionally to scene volume and only very weakly with the number of source voxels (compare the two sources in ZenGarden). We note some manual domain adjustment was performed for the two sources in Titanpass so their two domain sizes are not identical.

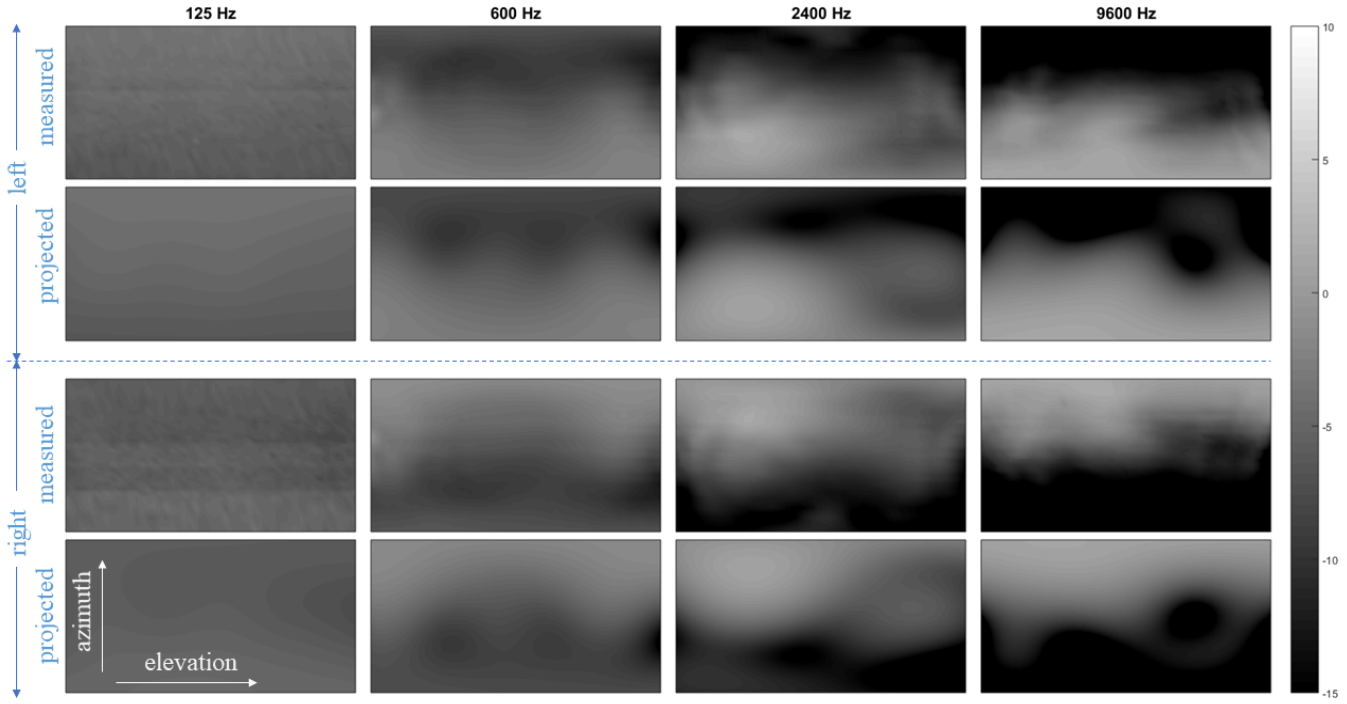

Fig. 6. HRTF directional power representation. We used subject #3 from the CIPIC database. First (left channel) and third (right channel) rows show average spectral power over each sub-band; second and fourth rows show the corresponding least-squares reconstruction using low-order SH. All fields are visualized in dB. The horizontal and vertical axes represent elevation (-45 to 235 degrees) and azimuthal angle (-80 to 80 degrees) using the interaural coordinate system [Algazi et al. 2001]. As expected, low-order SH reconstructs a smooth approximation of the measured data.

Runtime memory cost is around 1 MB per extended source and decoding cost about 100 microseconds per frame. HRTF-based rendering is also lightweight (transformation of the SH vector E from world to head space plus the dot product between the length-n^2 HRTF vector and the rotated SH vector E in each of the n_H sub-bands), making our framework immediately practical.

We demonstrate four scenes in the supplementary video: BeachHouse, Outpost23, TitanPass and ZenGarden. BeachHouse includes a single source in the form of a long cuboid representing the beach; the single source in Outpost23 comprises three large industrial fans. Two separate sources are included in TitanPass (waterfall and stream) and ZenGarden (rain falling into a water pool and rain falling everywhere else), allowing two different representative signals in each scene. The rain source is generated by tracing drops vertically from the ceiling of the simulation domain and placing an incoherent point where it first intersects scene geometry.

Figures 1 and 7 show parameter fields for ZenGarden and BeachHouse, respectively. They are smooth indoors and outdoors in scenes with complex portals and occluders. High-frequency spatial oscillation due to interference is avoided. Overall, our system is well spatialized, providing smooth loudness and directional cues from environmental propagation.

BeachHouse. This simple scene demonstrates shadowing and guiding of sound around the building and through its doors and windows. Inside the house, directionality indicates positions of the nearby portals. Outside, the ocean gets louder. Loudness variation is smooth and stays steady as the listener moves along the beach. Next to and behind the building, shadowing substantially reduces loudness while increased directionality can be heard at the sides and back corners of the building.

Outpost23. Loudness varies smoothly even when walking directly in front of the fans. Our directional rendering clearly points the way towards these sources as the listener navigates.

TitanPass. The waterfall is a big, loud source that dominates nearby. Directionality is clearly audible in the cavern by the falls. Descending past the lower falls, the burbling stream becomes the main sound. It is highly directional as heard from the bank and gets louder and more surrounding (isotropic) in the enclosed channel.

ZenGarden. A softer rainfall sound from a large area source covering the ground and rooftops permeates the scene, joined by splashing sounds of rain falling into the fish pond. Directionality towards the pool is clearly rendered while the background rainfall remains audible. Moving along the walkway next to the pool, rain sounds are shadowed smoothly by pillars supporting the roof.

Incoherent vs. coherent comparison. The video also compares the coherent point source technique of [Raghuvanshi and Snyder 2014] with our incoherent extended source method in the BeachHouse

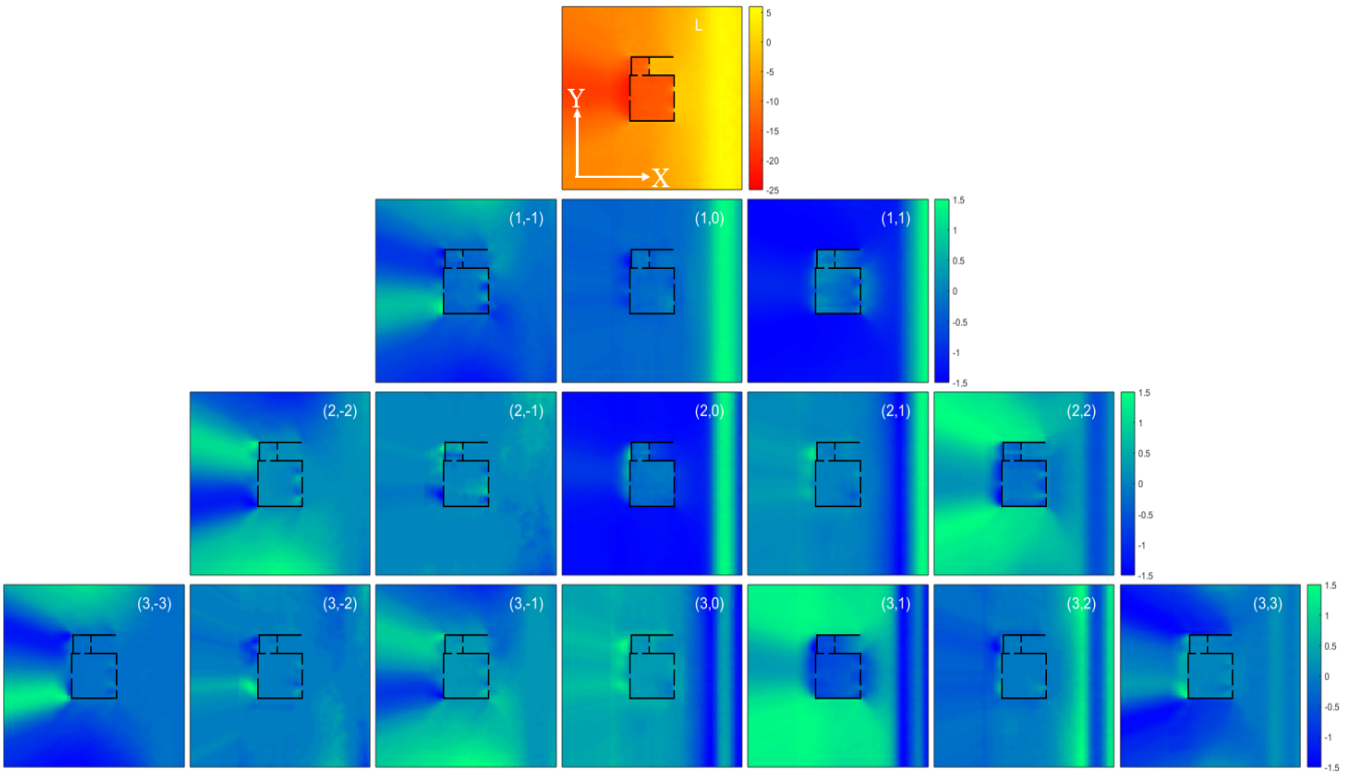

Fig. 7. Parameter fields in BeachHouse. Top row: total loudness field L in dB. Bottom rows: higher-order ($l > 0$) SH coefficients relative to DC, $E_{l,m}/E_{0,0}$, before windowing. Total loudness field L captures the spatial variation of ambient sound loudness and the high-order ($l > 0$) SH coefficients $E_{l,m}/E_{0,0}$ encode the directional power. All encoded parameter fields are spatially smooth.

scene. We placed nine point sources evenly along the coast where each emits a (coherent) Gaussian derivative pulse simultaneously in a single precomputed simulation. Because the nine sources are precomputed, runtime cost is similar to our technique; the cost would be significantly higher than ours if the sources were independently controlled at runtime. Several artifacts can be observed with this alternative. Moving along the ocean source, loudness wobbles unnaturally due to interference among the point sources. The shadow behind the house is also unrealistically sharp. Good results for large sources require numerous point sources (thousands in our experiments) with uncorrelated phase.

9 CONCLUSION

Assuming ideal incoherence of an ambient source over both time and spatial extent, we make it practical to render its propagated effects through a complex 3D scene. Unlike geometric methods in acoustics and CG, our method computes an Eulerian PDE simulation. Wave effects like diffraction are included and cost remains insensitive to scene complexity and source size. We show how the incoherent source signal can be evaluated efficiently and propose a streaming encoder to capture the time-averaged directional power distribution of the propagated response in terms of low-order spherical harmonics for each listener position. The resulting 3D parameter fields are smooth and compressible, needing just a few megabytes per source. Our system inexpensively generates convincing ambient effects that give salient information about the scene.

Many limitations remain to be addressed in future work. Frequency-dependent propagation effects are a straightforward extension which require extracting the relevant power in frequency bands from the response at each listener position and then performing the same analysis we propose for each band. More challenging extensions break our assumption of ideal incoherence, to capture near-field effects when sound events are individually audible, or add parameters that depend on the transient response for partially incoherent sources (e.g. delay/directionality for outdoor echoes).

Our directional rendering method can be improved. Its rationale is that with incoherent ambient sources, phases at the two ears are uncorrelated and only frequency-dependent shadowing effects are noticeable. We thus apply the same representative signal equalized at the two ears according to the listener's head shadowing effects, with matching phase. Decorrelating these phases [Valimaki et al. 2012] is more natural and would probably increase the feeling of envelopment.

Finally, Eulerian simulation could be applied to conventional light rendering to exploit its computational independence on scene complexity and source size. Since we don't expect it will be competitive to simulate big scenes at visible light wavelengths, such simulation will need to mitigate diffraction effects at longer wavelengths

Table 1. Precomputation data

scene/(source)	# scene voxels	# source voxels	scene surface area (m^2)	time steps	dimensions (m)	bake RAM (GB)	bake time (h)	encoded (MB)
BeachHouse	2.5×10^6	17.0×10^3	0.5×10^3	9.2×10^3	$45 \times 80 \times 8$	2.2	2.0	0.68
Outpost23	1.4×10^6	3.6×10^3	32.8×10^3	9.1×10^3	$40 \times 40 \times 10$	1.4	15.0	2.1
TitanPass (waterfall)	2.0×10^6	1.7×10^3	9.7×10^3	9.2×10^3	$20 \times 60 \times 21$	2.0	6.9	1.0
TitanPass (stream)	3.5×10^6	1.9×10^3	9.3×10^3	9.2×10^3	$20 \times 80 \times 28$	3.0	12.3	1.6
ZenGarden (rain-ground)	2.4×10^6	46×10^3	14×10^3	9.1×10^3	$50 \times 70 \times 8$	2.4	14.1	1.6
ZenGarden (rain-water)	2.4×10^6	4.0×10^3	14×10^3	9.1×10^3	$50 \times 70 \times 8$	2.4	13.8	1.6

and augment the wave simulation with BRDF/scattering models for more coarsely discretized geometry.

REFERENCES

V Ralph Algazi, Richard O Duda, Dennis M Thompson, and Carlos Avendano. 2001. The CIPIC HRTF database. In *Applications of Signal Processing to Audio and Acoustics, 2001 IEEE Workshop on the*. IEEE, 99–102.

Chunxiao Cao, Zhong Ren, Carl Schissler, Dinesh Manocha, and Kun Zhou. 2016. Interactive sound propagation with bidirectional path tracing. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 180.

Jeffrey N Chadwick, Steven S An, and Doug L James. 2009. Harmonic shells: a practical nonlinear sound model for near-rigid thin shells. *ACM Trans. Graph.* 28, 5 (2009), 119–1.

Robert L. Cook, Thomas Porter, and Loren Carpenter. 1984. Distributed Ray Tracing. In *Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '84)*. ACM, New York, NY, USA, 137–145. https://doi.org/10.1145/800031.808590

Brian Hamilton, Stefan Bilbao, Brian Hamilton, and Stefan Bilbao. 2017. FDTD Methods for 3-D Room Acoustics Simulation With High-Order Accuracy in Space and Time. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.* 25, 11 (Nov. 2017), 2112–2124. https://doi.org/10.1109/TASLP.2017.2744799

Jean-Marc Hasenfratz, Marc Lapierre, Nicolas Holzschuch, and François Sillion. 2003. A survey of Real-Time Soft Shadows Algorithms. *Computer Graphics Forum* 22, 4 (dec 2003), 753–774. https://doi.org/10.1111/j.1467-8659.2003.00722.x

IS ISO3382. 2009. Acoustics-Measurement of room acoustic parameters, Part 1: Performance spaces, ed. *B. Standards* (2009).

Doug L James, Jernej Barbič, and Dinesh K Pai. 2006. Precomputed acoustic transfer: output-sensitive, accurate sound generation for geometrically complex vibration sources. In *ACM Transactions on Graphics (TOG)*, Vol. 25. ACM, 987–995.

Jan Kautz, John Snyder, and Peter-Pike J Sloan. 2002. Fast Arbitrary BRDF Shading for Low-Frequency Lighting Using Spherical Harmonics. *Rendering Techniques* 2, 291-296 (2002), 1.

Konrad Kowalczyk and Maarten Van Walstijn. 2011. Room acoustics simulation using 3-D compact explicit FDTD schemes. *IEEE Transactions on Audio, Speech, and Language Processing* 19, 1 (2011), 34–46.

Mikko V. Laitinen, Tapani Pihlajamäki, Cumhur Erkut, and Ville Pulkki. 2012. Parametric Time-frequency Representation of Spatial Sound in Virtual Worlds. *ACM Trans. Appl. Percept.* 9, 2 (June 2012). https://doi.org/10.1145/2207216.2207219

Dingzeyu Li, Yun Fei, and Changxi Zheng. 2015. Interactive Acoustic Transfer Approximation for Modal Sound. *ACM Trans. Graph.* 35, 1 (Dec. 2015). https://doi.org/10.1145/2820612

Josh H McDermott, Michael Schemitsch, and Eero P Simoncelli. 2013. Summary statistics in auditory perception. *Nature neuroscience* 16, 4 (2013), 493.

Josh H McDermott and Eero P Simoncelli. 2011. Sound texture perception via statistics of the auditory periphery: evidence from sound synthesis. *Neuron* 71, 5 (2011), 926–940.

Ravish Mehra, Nikunj Raghuvanshi, Lakulish Antani, Anish Chandak, Sean Curtis, and Dinesh Manocha. 2013. Wave-based sound propagation in large open scenes using an equivalent source formulation. *ACM Transactions on Graphics (TOG)* 32, 2 (2013), 19.

Hans P Moravec. 1981. 3D graphics and the wave theory. *ACM SIGGRAPH computer graphics* 15, 3 (1981), 289–296.

Alan V Oppenheim. 1999. *Discrete-time signal processing*. Pearson Education India.

Nikunj Raghuvanshi and John Snyder. 2014. Parametric Wave Field Coding for Precomputed Sound Propagation. *ACM Transactions on Graphics (TOG)* 33, 4 (July 2014). https://doi.org/10.1145/2601097.2601184

Nikunj Raghuvanshi and John Snyder. 2018. Parametric Directional Coding for Precomputed Sound Propagation. *ACM Transactions on Graphics (TOG)* 37, 4 (Aug. 2018), 14.

Nikunj Raghuvanshi, John Snyder, Ravish Mehra, Ming C. Lin, and Naga K. Govindaraju. 2010. Precomputed Wave Simulation for Real-Time Sound Propagation of Dynamic Sources in Complex Scenes. *ACM Transactions on Graphics (proceedings of SIGGRAPH 2010)* 29, 3 (July 2010).

Yotka S Rickard, Natalia K Georgieva, and Wei-Ping Huang. 2003. Application and optimization of PML ABC for the 3-D wave equation in the time domain. *IEEE Transactions on Antennas and Propagation* 51, 2 (2003), 286–295.

Tobias Ritschel, Carsten Dachsbacher, Thorsten Grosch, and Jan Kautz. [n. d.]. The State of the Art in Interactive Global Illumination. *Computer Graphics Forum* 31, 1 ([n. d.]), 160–188. https://doi.org/10.1111/j.1467-8659.2012.02093.x

Lauri Savioja and U Peter Svensson. 2015. Overview of geometrical room acoustic modeling techniques. *The Journal of the Acoustical Society of America* 138, 2 (2015), 708–730.

Carl Schissler, Ravish Mehra, and Dinesh Manocha. 2014. High-order diffraction and diffuse reflections for interactive sound propagation in large environments. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 39.

Carl Schissler, Aaron Nicholls, and Ravish Mehra. 2016. Efficient HRTF-based spatial audio for area and volumetric sources. *IEEE transactions on visualization and computer graphics* 22, 4 (2016), 1356–1366.

John B Schneider and Christopher L Wagner. 1999. FDTD dispersion revisited: Faster-than-light propagation. *IEEE Microwave and Guided Wave Letters* 9, 2 (1999), 54–56.

Dirk Schröder. 2011. *Physically Based Real-Time Auralization of Interactive Virtual Environments*. Logos Verlag. http://www.worldcat.org/isbn/3832530312

Peter-Pike Sloan. 2013. Efficient spherical harmonic evaluation. *Journal of Computer Graphics Techniques* 2, 2 (2013), 84–90.

Peter-Pike Sloan, Jesse Hall, John Hart, and John Snyder. 2003. Clustered principal components for precomputed radiance transfer. In *ACM Transactions on Graphics (TOG)*, Vol. 22. ACM, 382–391.

Peter-Pike Sloan, Jan Kautz, and John Snyder. 2002. Precomputed Radiance Transfer for Real-time Rendering in Dynamic, Low-frequency Lighting Environments. *ACM Trans. Graph.* 21, 3 (July 2002), 527–536. https://doi.org/10.1145/566654.566612

Julius Orion Smith. 2008. *Introduction to digital filters: with audio applications*. Vol. 2. Julius Smith.

Allen Taflove and Susan C Hagness. 2005. *Computational electrodynamics: the finite-difference time-domain method*. Artech house.

Vesa Valimaki, Julian D Parker, Lauri Savioja, Julius O Smith, and Jonathan S Abel. 2012. Fifty years of artificial reverberation. *IEEE Transactions on Audio, Speech, and Language Processing* 20, 5 (2012), 1421–1448.