Adaptive Sampling for Sound Propagation

Chakravarty R. Alla Chaitanya, John M. Snyder, Keith Godin, Derek Nowrouzezahrai, and Nikunj Raghuvanshi

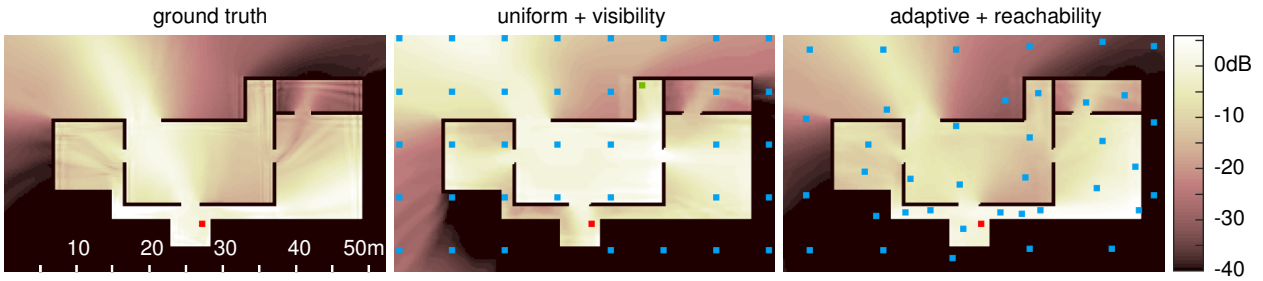

Fig. 1. Scene-aware sampling and interpolation. The left image shows the ground truth direct loudness field in the HOUSE scene for a runtime listener shown by the red point. (All images show a 2D slice of the 3D set of possible runtime source locations.) Middle image shows the field reconstructed using prior work via uniform probe sampling and visibility-based interpolation. Right image shows the more faithful match obtained by our new methods for adaptive sampling and reachability-based interpolation. Uniform sampling generates large errors in the central room and to the lower left of the scene. Probe samples are shown as blue points. The green point in the middle column shows a failure case for visibility interpolation: no probes are directly visible to a player located there and decoding fails entirely. On a similarity score ranging from 0 to 1 detailed in Sect. 8, where 1 is perfect match to ground truth, our work improves the match to 0.93 from 0.83 with prior art.

Abstract—Precomputed sound propagation samples acoustics at discrete scene probe positions to support dynamic listener locations. An offline 3D numerical simulation is performed at each probe and the resulting field is encoded for runtime rendering with dynamic sources. Prior work place probes on a uniform grid, requiring high density to resolve narrow spaces. Our adaptive sampling approach varies probe density based on a novel "local diameter" measure of the space surrounding a given point, evaluated by stochastically tracing paths in the scene. We apply this measure to layout probes so as to smoothly adapt resolution and eliminate undersampling in corners, narrow corridors and stairways, while coarsening appropriately in more open areas. Coupled with a new runtime interpolator based on radial weights over geodesic paths, we achieve smooth acoustic effects that respect scene boundaries as *both* the source or listener move, unlike existing visibility-based solutions. We consistently demonstrate quality improvement over prior work at fixed cost.

Index Terms—Diffraction, interpolation, mean free path, radial basis function, ray tracing, reciprocity, room acoustics, wave simulation

1 INTRODUCTION

Games and virtual reality (VR) require efficient and convincing simulations of real-world phenomena in order provide users with an uninterrupted, immersive experience. In particular, sound propagation effects that take the scene geometry into account provide important perceptual cues about the location of sound sources in the environment, as well as the environment itself. Imagine, for example, walking away from a crowded room in a party, down a corridor and around a corner. The manner in which the sound intensity diminishes depends not only on the distance from the crowd, but also on complex secondary effects like the diffraction and reflection of sound around doorways and corners. These subtle, but important, aural cues are essential to provide a listener with a convincing experience.

Unfortunately, such global sound transport effects are costly to simulate, requiring extensive computation to evaluate integrals of complex sound paths through the scene that link sources to listeners. As such, brute-force computation is not a feasible solution for interactive simulation, where performance budgets are prohibitively constraining. This motivates precomputation-based solutions, such as existing systems for lighting [9] and sound propagation [14, 15]. These systems perform the expensive transport simulation offline at a sampled set of *probe* locations, whose results are interpolated to the player location at runtime. Propagated energy distribution in complex scenes is a piecewise continuous function with arbitrarily large physical discontinuities across boundaries like walls. Two problems must be solved to accurately reconstruct such functions: the number of probes must be economized while ensuring narrow regions in the scene such as corridors remain adequately sampled, and the interpolation must retain accuracy inside each smooth region and preserve differences across boundaries.

We focus on precomputed sound propagation, building upon parametric wave coding [14] as a case study. Similar to other precomputed systems [10, 12, 16], it employs uniform probe sampling. Density must be increased globally to sample thin regions (1-2m across), requiring impractical precomputation time and runtime RAM usage while wasting samples in open areas with low variation. At runtime, precomputed probe information must be interpolated at the listener location, usually using linear interpolation. To avoid incorrectly blending jumps in the field across walls, probes invisible to the listener are rejected but this introduces artificial jumps at visibility transitions. Human auditory perception is sensitive to such unphysical loudness changes on slight player motion, breaking immersion in a VR experience. We make two main contributions to fix these problems.

First, we propose a novel geometric measure of the *local diameter* of the scene around an input point that enables scene-adaptive sampling. Local diameter captures a notion of the scene's spaciousness or narrowness around a given point, defined via stochastic path tracing.

- *C. R. Alla Chaitanya is with Microsoft Research and McGill University. E-mail: chakravarty.alla@gmail.com.*
- *J. M. Snyder, K. Godin and N. Raghuvanshi are with Microsoft Research. E-mails: {johnsny, kegodin, nikunjr}@microsoft.com.*
- *D. Nowrouzezahrai is with McGill University. E-mail: derek@cim.mcgill.ca.*

Manuscript received 10 Sept. 2018; accepted 7 Feb. 2019. Date of publication 17 Feb. 2019; date of current version 27 Mar. 2019. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TVCG.2019.2898765

1077-2626 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

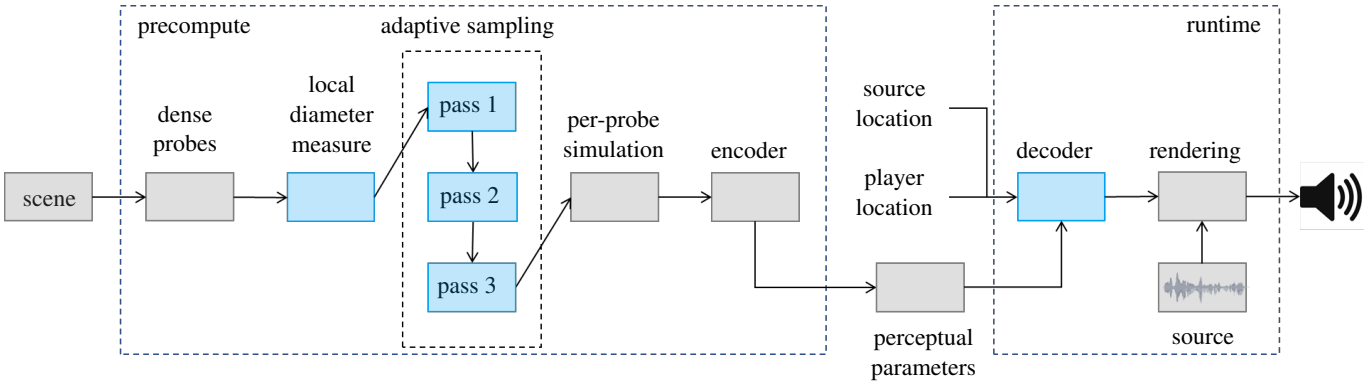

Fig. 2. Overview of the system. We build atop a parametric wave coding model [14], highlighting our modifications (blue boxes). We replace uniform sampling with adaptive sampling in the precomputation stage to improve sampling of potential player positions (probes). At runtime, the actual, dynamic player location applies our reachability-based interpolation inside the decoder to improve the accuracy of interactive audio rendering.

Controlled by a single locality parameter, our measure progresses between mean diameter of the region visible to the point on one extreme, to the global mean free path distance in all parts of the scene accessible from the point. The former is too sensitive to local geometry, diminishing in corners of a large room, while the latter is too global, linking across arbitrarily many connected spaces. We obtain an effective balance with intuitive variation. Our sampling approach then starts with a dense layout and greedily merges probes closer than the local diameter, yielding an unstructured set of probes whose density adapts to local spaciousness.

Second, we complement this unstructured adaptive sampling with a reachability-based interpolation technique that affords reconstruction accuracy and smoothness. It selects probes near the player and applies a radial basis function (RBF) weighting based on a heuristic diffraction model, namely, the ratio of Euclidean to geodesic shortest path distance. Weight thus diminishes smoothly as energy flow from a probe sample to the player attenuates rather than cutting out abruptly on visibility transitions, even functioning when no probes are visible (e.g., green point in middle panel of Fig. 1).

These enhancements substantially improve accuracy of precomputed approaches such as parametric wave coding [14] for the same probe budget, as shown in Fig. 1. Our system is the first to capture sound propagation effects in scenes with arbitrarily thin walkable regions meeting a practical budget for precomputation time and runtime resources (10% of a single CPU core and about 100MB of RAM for industrial-calibre game environments). The technique has been successfully employed in a shipping game [17].

2 Related Work

We review prior art in the areas most related to our work, below.

Propagation solvers and online approaches Geometric solvers are commonly used in room acoustics [8], similar to global illumination techniques but with two differences. First, time delays are audible for sound. Energy must be accumulated in multiple time bins, making path length an additional sampling dimension. Second, ray propagation assumes high frequency approximation to the wave equation. Diffraction approximations for audible wavelengths must be explicitly incorporated by sampling edges. Both differences increase the computational demands of path sampling.

The acoustic response can be computed on the fly between a dynamic source and listener for simple scenes of a few thousand polygons [21], while saturating multiple desktop PCs. Other techniques use path caching with enforced smoothness for plausible effects in complex game scenes [20]. Controlling aliasing without breaking the CPU budget remains a challenge [2]. Modeling arbitrary order wave diffraction and scattering poses additional difficulties; see Savioja et al. [19] for a survey and discussion of geometric solvers.

Wave solvers [4, 13] evolve the volumetric pressure field in a discretized approximation to the wave equation. Aliasing is eliminated simply by proper Nyquist bandlimiting of the source pulse. These techniques are too slow to evaluate in realtime and produce petabyte-sized raw output.

Precomputed approaches Precomputed approaches run a propagation solver on static scene geometry offline, trading increased memory for reduced runtime computation. Ensuring the sampling obtains an accurate reconstruction while minimizing the number of samples has been largely unaddressed and forms our focus.

An early approach [12] assumes stationary sources and proposes a uniformly spaced layout of listener probes in simple, convex scenes. Tsingos [25] extends this framework to enable one of the first practical techniques for games and VR. Diffraction is ignored and input geometry limited to a few planar facets, but dynamic sources are enabled and the data size reduced substantially using image sources. Probes are placed by hand in a visual authoring tool. Our work considers complex games scenes where hundreds or thousands of probes are needed per game map, making manual placement impractical.

Raghuvanshi et al. [16] use a wave solver to accommodate complex scene geometry and naturally model diffraction but the representation requires gigabytes of memory. Raghuvanshi and Snyder [14] reduce data size to about 100MB, practical for current games, by encoding acoustic impulse responses in terms of perceptual parameters. We discuss relevant details in Section 3. Both papers share the sampling framework, laying out probes on a uniform grid and performing an expensive 3D wave simulation from each probe location, whose output data is encoded and stored. To control cost in large game scenes, coarse spacing is used, typically 3-4m, risking inaccuracies in narrow regions.

The equivalent source method (ESM) fits sound fields using a linear superposition of elementary multipoles. James et al. [7] employ ESM to approximate free field radiation from vibrating objects. Multipole placement is unstructured and greedily optimized to fit the known radiating field. The wave field we're evaluating is unknown at the probe layout stage, ruling out a fitting approach. In principle, one could compute 3D wave fields over a dense set of probes and then decimate the set as an optimization. This is accurate but impractically expensive and wasteful. We seek a fast geometric measure to adapt probe spacing before computing the costly simulations.

Mehra et al. [10] use a similar framework as parametric wave coding [14]: probes are first laid out with sparse uniform spacing and a 3D simulation performed for each; it is the encoding of each probe's 3D field that differs. Assuming open, outdoor scenes with a few well-separated objects, the field is approximated using ESM as a superposition of multipole radiation from each object.

Taylor et al. [24] use a heuristic similarity measure based on geometric acoustics in the scene, combined with visibility criteria. Acoustic-based sampling heuristics only work reliably to the extent they match

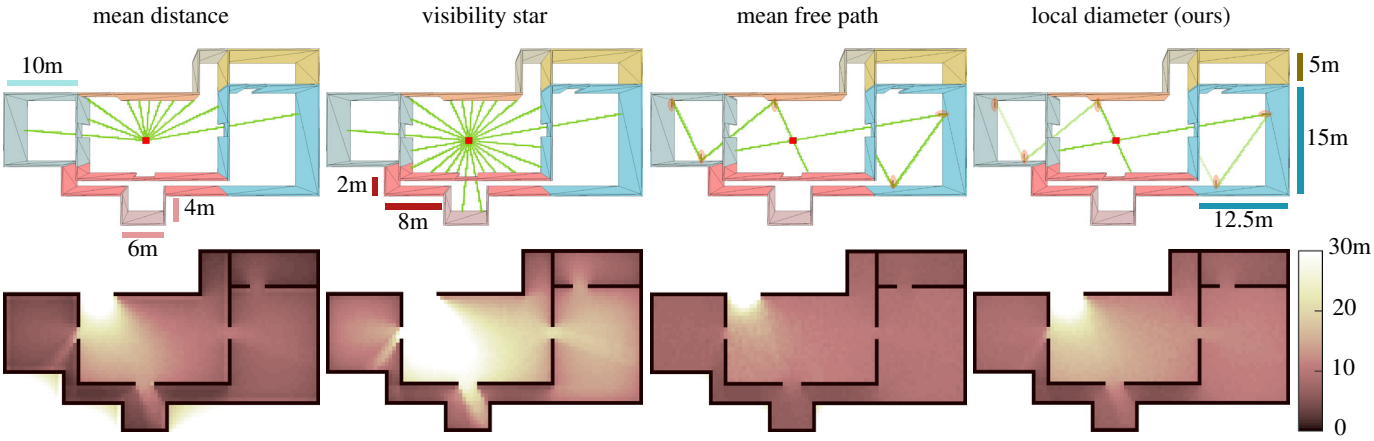

Fig. 3. Local diameter measure – The top row illustrates computation of various diameter measures at a point shown by the red dot. We only show a few example paths in green. The bottom row shows the corresponding result on repeating the diameter evaluation throughout a test scene. *Mean distance* traces rays forward from the evaluation point and records the average distance to intersected geometry. *Visibility star diameter* adds the distance from shooting the ray in the opposite direction to reduce the bias towards short path segments for proximate geometry. *Mean free path* performs diffuse scattering off intersected geometry, avoiding sensitivity to corner proximity at the cost of being too global, with limited variation between rooms. *Local diameter* is our proposed measure which multiplicatively attenuates the weight for a path segment by a factor λ at each bounce (illustrated in image with segment fadeout), parameterizing locality of the transport space.

the offline simulation. We propose a simple and fast approach that instead tries to sample locations listeners can reach. Combined with our interpolation approach, this resolves geometry-related jumps in the acoustic field and we show that it works well in complex scenes.

We build upon parametric wave coding [14] to quantify improvement, but note that precomputed acoustic systems differ primarily in how they compute and encode their data, but share sampling tradeoffs. Our techniques are thus applicable to other systems.

Light probes Image-based lighting (IBL) captures light probes, represented by cubemaps [6, 9] or spherical harmonics [18, 23], to provide realistic shading including indirect effects at runtime. Refer to these talks from Debevec [3] and O'Donnell [11] for more extensive background information. Probes are placed by hand in most applications; automatic placement that avoids incorrect shading and light leaking [6] is a long-standing research problem. McGuire et al. [9] calculate real-time global illumination in static environments using a data structure that provides a spherical slice of the light field, surface normals, and radial distances at any point based on a uniform grid of probes. Silvennoinen et al. [22] reconstruct incident radiance from probes that are decimated from an initial overcomplete set to one containing a specified number. Output sampling density remains roughly uniform. Like ours, their technique uses RBFs for interpolation but based on Euclidean distance rather than geodesic distance respecting scene geometry, a more suitable model for sound transport. Our local diameter measure potentially offers improved light probe placement that can adapt to the scene.

3 Background

Summarizing parametric wave coding [14], acoustic precomputation is a 7D problem as both source and listener can move in 3D, and a time-dependent impulse response results for each pair. Dimensionality is reduced by observing that the space of player/listener locations is usually 2.5D, restricted to walkable surfaces and available as the "navigation mesh" in game engines. Reciprocity allows swapping of the source and listener without affecting the acoustic response; the more restricted player is thus treated as the probe during precomputation. Each simulation for an impulse placed at a probe location yields an emitter field representing the acoustic response for a (runtime) source free to move in 3D, as heard by the player located at that probe.

Each impulse response is encoded as four scalar parameters that perceptually characterize it: the direct sound loudness (dB) of initial energy arriving at the listener during the first 10ms, L_{DS}; the loudness of early reflections in the following 200ms, L_{ER}; the decay time of reflections, T_{ER}; and the decay time of late reverberation, T_{LR}. The overall encoded data is a set of 3D emitter fields for these four parameters, for each probe location. Each emitter field is sub-sampled uniformly at a relatively fine spacing of around 1m and compressed by the encoder. Probe layout must be considerably sparser at 3-4m (see parametric wave coding [14, Table 2]) to keep precomputation and RAM costs in check, which increase quadratically as probe density increases.

We note that L_{DS} is most spatially variable and spatially sensitive to surrounding geometry because it integrates over the shortest time window. It exhibits interesting wave behavior including (soft) diffracted shadow edges and (limited) interference oscillations. This motivates its use as our primary test parameter for reconstruction errors. To isolate errors introduced by probe sampling, we eschew sub-sampling for the emitter fields, instead storing the dense 3D field for each probe at the wave solver's spacing of 25.5cm.

Fig. 2 overviews our system. During precomputation, we generate a dense set of probe locations above the scene's navigation mesh (Sect. 6). We develop a geometric measure of the local diameter of space (Sect. 4) and use it to prune probe samples adaptively (Sect. 5). We then perform a 3D numerical wave simulation at each probe location and encode impulse responses between source and listener locations in terms of the four aforementioned perceptual parameters. At runtime, we apply reachability-based interpolation (Sect. 7) to the decoded parameters to filter each source sound, yielding smooth and realistic audio as sources and listeners move through the scene.

4 Local Diameter Measure

To control probe density, we measure local diameter, $d(x)$, as a function of scene geometry and an evaluation point x in the scene. Refer to Fig. 3. Our measure should satisfy several properties. It should correspond to the space surrounding x, shrinking in narrow corridors and growing in open spaces. Within a single chamber it should be roughly constant, but also vary smoothly as x moves between two spaces connected by a portal. This implies the measure can be neither purely local to the geometry nearest x, nor a global summary of the entire scene.

Perhaps the simplest such measure is mean distance to geometry around x, shown in the left column of the Fig. 3. It is very sensitive to geometric proximity. Some improvement can be obtained by instead computing the "visibility star diameter" around the point (second column), by tracing rays in opposite directions around x and combining their resulting distances. This still exhibits undesirably large reduction near corners, variation within rooms such as the central hall in the

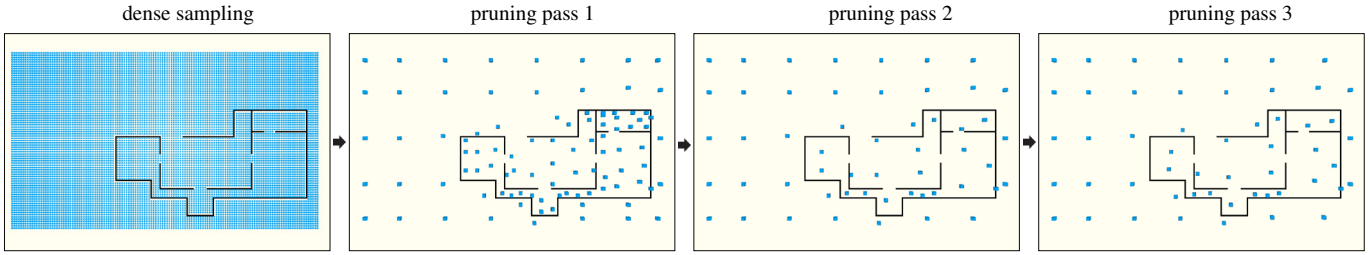

Fig. 4. Illustration of pruning passes in our adaptive sampling approach. We start with an initial densely sampled set of probes (left) that ensure sampling in all areas of the scene. The probes are inserted into a grid and greedy merges are performed within each grid cell based on the local diameter measure at each probe point. This results in an initial adaptive sampling (second from left). Merges across grid cells are then performed (third from left). Finally, probes in small, completely isolated regions of space (such as inside pillars) are removed. This has no effect in the case illustrated (rightmost).

figure, and leakage from large spaces into small ones through portals. In short, it is too local.

To make the measure more global, we consider mean free path length. Sabine's classic analysis for reverberation time under stochastic "diffuse field" conditions [8] builds on the theorem that the mean free path in a scene is proportional to V/A, where V is its volume and A its surface area. For convex scenes, V/A represents a measure of average scene diameter. We generalize this idea via stochastic path tracing. Paths are traced emanating from x and terminating at a distance of 100m. At each bounce (i.e., intersection with scene geometry), the path is redirected in a uniformly random direction around the surface normal hemisphere, and the average length over all path segments computed. The pair of initial segments originating from the point in opposite directions is summed as in the visibility star measure, so that segment distances always correspond to rays bounded on both sides by a geometric intersection. The result (third column) is indeed smoother and more constant in each room but is also more constant across the entire scene as paths escape from small rooms and generate segments that bounce around in much larger connected spaces.

Our proposed measure strikes a parameterized balance between these two extremes by performing a weighted mean of segment lengths, where the weight at each bounce is reduced by some factor. Segment weight thus decreases exponentially with bounce order. The factor, λ, ranges between a purely local measure to an entirely global one as λ varies in $[0,1]$. More precisely, $\lambda = 0$ yields the local visibility star measure while $\lambda = 1$ yields the global mean free path. Choosing $\lambda = 0.8$ provides a good balance, as shown in the rightmost column of Fig. 3. Compared to the mean free path, the corridors darken and are less influenced by the connected large hall. The central hall brightens, bringing its value closer to its actual width neglecting portals. We observe similar behavior across our test scenes. Namely, results remain smooth at the connection between small and large enclosures, with diameter values that correspond to each individual space. To demonstrate this, we set $\lambda = 0.8$ for all our results.

Our measure performs scattering to ensure spatial smoothness and make it more global, not to match expected acoustic response. We compute local diameters in 2D at a fixed height from navigable floors (further detail is included in Sect. 6). We have found this sufficient for our examples; a 3D measure is straightforward but more expensive.

Voxelization tracing Though other implementations are certainly possible, we compute the geometric intersection and scattering needed for $d(x)$ evaluation using a scene voxelization rather than tracing rays directly in the original triangulated scene. (This voxelization is the same one used for simulating wave acoustics.) The computation thus has no knowledge of polygon normals around which to form a precise directional distribution for diffuse scattering. We select the normal of the first face of the occupied voxel hit by the ray. We construct an orthonormal basis around the normal and sample a new direction from a cosine- weighted hemispherical distribution. Since we compute local diameter in 2D, we project this reflection direction from 3D onto the $z = 0$ plane. Rays reaching the limits of the scene are terminated there as if reaching a boundary, without performing any scattering.

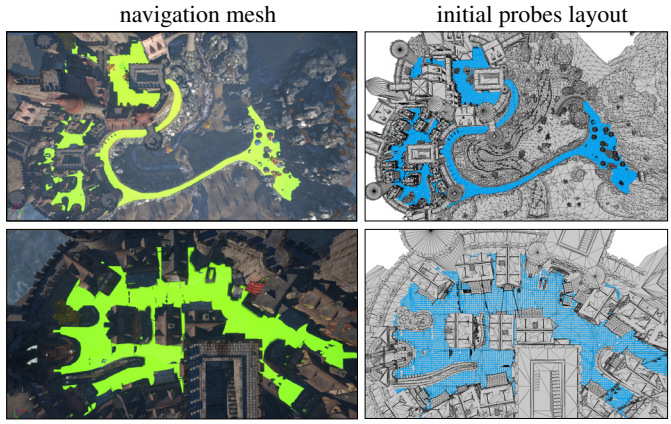

Fig. 5. EPICCITADEL navigation mesh (green) used to generate initial dense probes (blue).

5 ADAPTIVE SAMPLING

We start with a dense, uniformly sampled set of probes $\{P_0\}$, restricted to the 2.5D surface at human height above walkable areas of the scene. We provide details in Section 6. Sample spacing is 0.5m, fine enough to resolve narrow regions. Our approach then prunes probes from this set based on the measure $d(x)$, using the passes illustrated in Fig. 4.

Pruning predicate The user specifies the maximum allowable probe spacing, d^*, and a relative density parameter, s, that controls the probe sample density (per diameter). We use $d^* = 10$m in our tests, and $s = 2$ unless otherwise specified. Pruning relies on the predicate

$$\mathscr{P}(x_0,x_1) \equiv \left(\|x_0 - x_1\| < \frac{\min(d(x_0), d(x_1), d^*)}{s} \right) \wedge \mathscr{R}(x_0,x_1) \quad (1)$$

where $\mathscr{R}$ is true when the two points are mutually reachable via some path through the scene. $\mathscr{P}(x_0,x_1) = 1$ if x_0 and x_1 are close enough to merge. To evaluate $\mathscr{R}$, we voxelize geometry onto a grid with user-specified spacing (same as wave solver's grid spacing in our tests) and apply a flood fill, restricted to an appropriate scene sub-volume to accelerate the computation as we will later note in each case.

Pruning can be stated formally as finding a set of representatives $\{x_i\} \subset \{P_0\}$ in which all pairs violate the predicate, $\mathscr{P}(x_i,x_j) = 0$, but cover the scene; i.e., for all x in the scene, $\mathscr{P}(x,x_i) = 1$ for at least one representative x_i. This is a combinatoric optimization problem with $O(10^4)$ points in $\{P_0\}$, motivating our multi-pass greedy approach.

Pass 1: structured pruning We overlay a grid with spacing d^* onto the scene and partition probes within each grid cell i, yielding $P_0^i \subset P_0$. We then prune each P_0^i independently to a smaller set P_1^i,

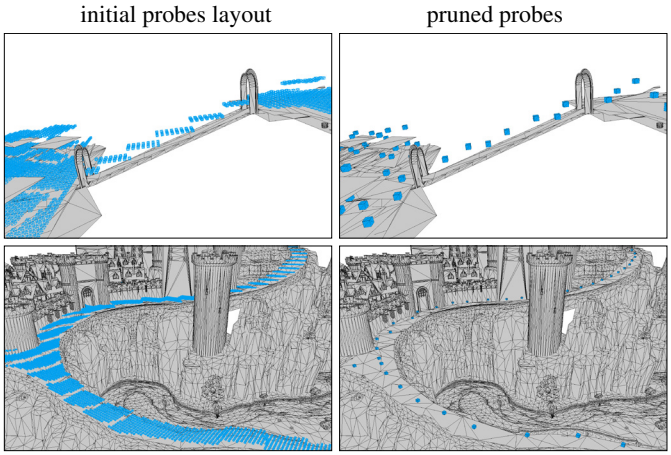

Fig. 6. Cross-sectional views of EPICCITADEL scene showing initial dense and pruned probes (in blue) at different z slices.

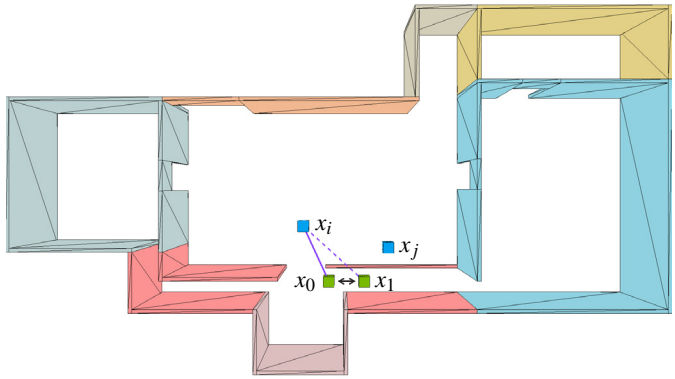

Fig. 7. Visibility-based interpolation yields non-physical discontinuities. As the player moves between x_0 and x_1, the contribution of probe x_i switches abruptly causing audible loudness variation. Our reachability-based interpolation ensures smoothness in such cases, while correctly ensuring that harder-to-reach probes (e.g., x_j) receive much lower weight.

initialized to empty. We pick an arbitrary probe $x_0 \in P_0^i$ and remove it along with all $x_1 \in P_0^i$ such that $\mathscr{P}(x_0, x_1) = 1$. (The sub-volume used for evaluating $\mathscr{R}$ in this predicate is the grid cell.) Call this set of pruned probes Q. We find the probe position $y \in Q$ closest to its centroid, and add the tuple (y, n) to P_1^i, where $n = |Q|$. The probe located at y thus serves as a representative for the set Q deemed close enough to the chosen x_0. The set size n is recorded to provide the weight when merging probes in the second pass. This procedure is repeated until P_0^i is empty, yielding $P_1 \equiv \cup_i P_1^i$.

Pass 2: unstructured pruning Pass 1 fails to prune probes straddling the artificial grid cell boundaries that might satisfy the predicate. We perform a second unstructured pass to remove such probes close to cell boundaries. We initialize a new pruned set $P_2 \leftarrow P_1$ followed by in-place merge operations. We find a pair $((y_0, n_0), (y_1, n_1)) \in P_2$, which satisfies the following:

- (y_0, y_1) must be in the same or spatially adjacent (including diagonals) grid cells. This test accelerates computation by observing that for probes in cells beyond immediate grid neighbors $\|y_0 - y_1\| > d^*$, violating $\mathscr{P}$.
- If (y_0, y_1) are in neighboring cells, at least one probe must be close to the grid cell edge, defined as being within distance b to the cell's edge. We use $b = 1$m.
- $\mathscr{P}(y_0, y_1) = 1$ with $\mathscr{R}$ evaluated on the axis-aligned bounding box formed by (y_0, y_1).
- Compute the weighted centroid $z = (n_0 y_0 + n_1 y_1)/(n_0 + n_1)$. This takes into account the number of initial dense probes in P_0 that y_0 and y_1 represent.
- z must be in air, not lying inside geometry.

If all tests pass, remove both probes from P_2 and replace them with the tuple $(z, n_0 + n_1)$. Repeat until no such pair can be found.

Pass 3: island culling Finally, a global flood fill is performed for each probe $z \in P_2$, and if the connected volume around z is below some lower bound $V_{\min} = 125\text{m}^3$, we remove z. This yields the final probe set, P.

6 GENERATING A DENSE PROBE SET USING THE NAVMESH

Commonly used game engines like Unreal and Unity allow automatic generation of a *navigation mesh* or *navmesh* for short. It is a 2.5D triangulated mesh indicating walkable regions in the map used to help game AI characters navigate. The user specifies geometry in the map that participates in the navmesh computation, such as terrain and static mesh objects within the playable region. A flood-fill along horizontal or inclined but sufficiently flat surfaces is then performed. This results in a 2.5D mesh that, for instance, climbs up stairwells so as to connect multiple floors in a building. Since acoustic probes must be placed in regions accessible by the player, we observe that the navmesh conveniently provides information to generate the initial dense set of proposed probe locations. Fig. 5 shows the EPICCITADEL navmesh (in green) we used for generating our results. The initial dense probes (in blue) are laid out using the navmesh as a guide. The user can further specify regions of interest by drawing boxes in the game editor which further restrict the probe set to the interior of their union.

We begin by voxelizing the navmesh at the same grid resolution used for all other processing described above. Then we consider a uniform sampling at 0.5m of the scene's voxel map, rejecting samples occupied by geometry or outside the region of interest. Assuming a player height of 2m, the navigation mesh voxel map is accessed in the interval of [1,2] meters below the sample. If any voxel occupied by the navigation mesh is detected, that sample is added to the dense probe set; otherwise it is rejected. This yields the initial dense probe set. The procedure naturally quantizes probe sample height into 0.5m steps. We process each occupied height slice independently as described in Section 5. Although our procedure introduces height "stair-casing" it is sufficient to handle challenging cases such as inclined tunnels and terrain as shown in Fig. 6.

7 RUNTIME INTERPOLATION

The runtime system interpolates the precomputed data , which encodes impulse responses between source and listener locations, in terms of a few perceptual parameters [14]. We interpolate results at receiver samples near the emitter, for each probe sample from a sample set near the listener. Receiver samples are on a uniform grid and are interpolated with trilinear weights, renormalized to account for missing or invisible samples, same as parametric wave coding [14]. However, we cannot use trilinear weighting for probe interpolation, because it requires uniform spacing of probe samples. Also, at visibility transitions the weighting is discontinuous and results in audible jumps. We propose a scheme that accommodates unstructured probe sample locations, and replace the discontinuous visibility criterion with a diffraction heuristic that yields smoother results.

Weight computation Given listener position x, we first find the nearest K probes, with locations $\{x_i\}$ and corresponding RBF weights $w_i = 1/|x - x_i|$. For each probe, we compute the geodesic shortest path length $g(x, x_i)$ connecting the listener to probe via the 3D scene (using A* search on a voxel grid, detailed shortly). If no connecting path is found, $g(x, x_i) = \infty$. We then compute a heuristic diffracted attenuation coefficient α_i by passing the "path directness" function $\rho(x, x_i) \equiv |x - x_i|/g(x, x_i)$ through the linear mapping $[\rho_m, 1] \rightarrow [0, 1]$

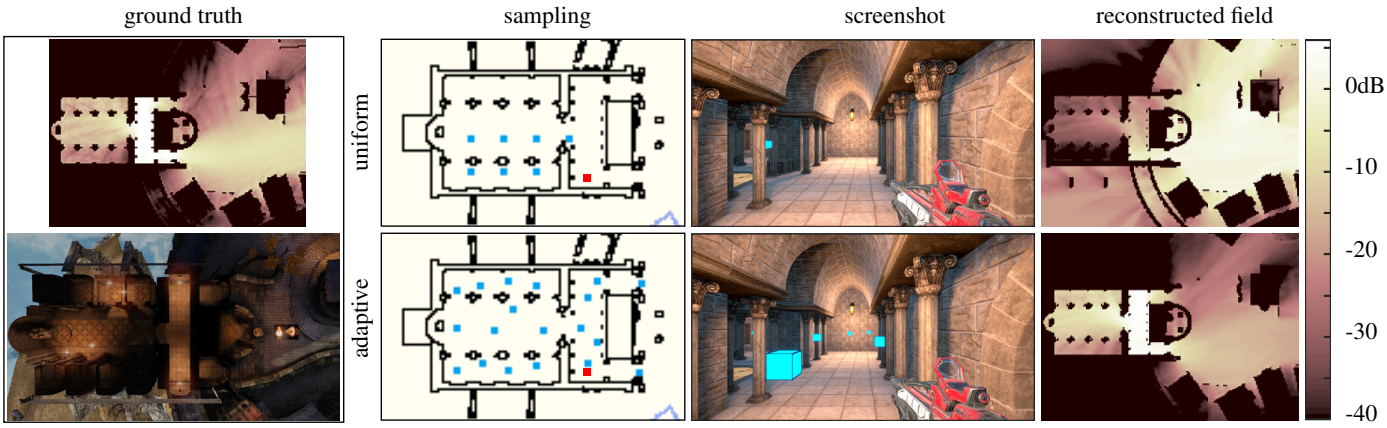

Fig. 8. Comparison on EPICCITADEL; probes in blue, listerner in red. Uniform sampling (top) can place probes in voxels occupied by geometry, undersampling areas like the top half of the cathedral. Shifting such samples to neighboring unoccupied voxels does not provide sampling guarantees invalidates the structured sampling required for linear interpolation. Our technique (bottom) ensures sufficient sampling. Paired with unstructured interpolation, runtime rendering matches ground truth (compare upper left image to right column).

with output clamped to $[0,1]$. We update the weights via $w_i \leftarrow \alpha_i w_i$, discard candidates with 0 weight, and re-normalize. This results in probe interpolation weights for the listener. One must also perform interpolation at the sound source location, which remains the same as parametric wave coding [14].

The proposed weight computation is robust to geometric complexity. Attenuating based on path deviation mimics smooth diffracted occlusion. For instance, tiny geometry like a leaf can block the segment (x,x_i) but unblock upon the slightest player motion, creating audible discontinuities not present in the encoded field. As shown in Fig. 7 such cases also arise when the player at x_0 is visible via a portal to a probe at x_i but becomes invisible on slight motion to position x_1. With our method, path directness ρ stays close to 1 in such cases so that the weight adjustment yields $\alpha_i \approx 1$, avoiding such discontinuities. Yet we preserve physical discontinuities due to scene boundaries: although a player at x_1 is close to probe x_j in the figure, the geodesic path takes a long detour via the portal resulting in small path directness satisfying $\rho < \rho_m$ and causing the probe x_j to be correctly discarded.

With the design decisions and acceleration techniques described next, the interpolation remains lightweight, requiring on the order of a few tens of microseconds for decoding the acoustics between a source and listener on a single CPU core.

Grid spacing Geodesic path computation on triangle meshes in complex game scenes can be challenging. Similar to the precomputation stage, we voxelize scene geometry onto a grid with user-specified resolution, h, on which we compute $g(x,x_i)$ approximately using A* search. This grid spacing must be chosen carefully. Reducing it improves accuracy but increases CPU cost and voxel memory usage as $1/h^3$. Another consideration is that if x lies within a grid cell occupied by a scene triangle, such as when the listener's head is right next to a wall, interpolation will fail. Fixing $h = 8$cm, about the typical radius of a human head, works well in practice.

Acceleration techniques To accelerate the geodesic path computation, we first test for the common case that x and x_i are visible to each other using a ray cast and in such a case, return $g(x,x_i) = |x-x_i|$.

To limit path search volume, we observe that no paths longer than $|x-x_i|/\rho_m$ need be explored. Using the triangle inequality, path search can be limited to points y such that $|y-x|+|y-x_i| \leq |x-x_i|/\rho_m$. This represents the interior of an ellipsoid with x and x_i as its foci. Thus we limit path search to the axis-aligned bounding box of this ellipsoid. Interpolation values are also cached and reused when the listener is stationary or moving slowly.

The original A* algorithm [5] proceeds by inserting candidate neighboring voxels into a priority queue, with priority drawn from a heuristic. We use Euclidean distance as this heuristic. We also perform some simple accelerations that sacrifice A* optimality. Instead of adding the unoccupied immediate neighbors of a voxel to the A* priority queue, we consider unoccupied voxels that are r voxels away in the directions of the immediate neighbors. All such candidates for which the straight line path is unblocked are added to the priority queue. We use a small value of $r = 3$, corresponding to hops of $r \times h = 24$cm that accelerates without undue inaccuracy. Once this search finds a cell within r voxels to the destination, the path is completed with a straight line if unblocked. Otherwise, A* search is performed with $r = 1$ to complete the remaining path.

We interpolate data from the $K = 4$ nearest probes to the listener, motivated by the 2D nature of our interpolation. Larger K increase computation without improving quality much. We fix $\rho_m = 0.5$ so that the weight coefficient vanishes ($\alpha_i = 0$) for geodesic paths longer than twice the Euclidean distance. Smaller values of ρ_m can help to find more complex paths connecting the listener to nearby probes, at the cost of performance as it requires a larger volume during A* search. It can also hurt quality, by giving weight to probes that are acoustically isolated from the player location. A value of $\rho_m = 1$ reduces to visibility based interpolation. Choosing $\rho_m = 0.5$ empirically yields good quality while limiting the computational cost of interpolation.

8 RESULTS

Existing precomputed sound propagation methods [10, 12, 14, 16], including commercial systems [1], propose uniform sampling. Some also include visibility-based interpolation. This prompts our use of uniform sampling as a baseline for comparison. While the light probe placement literature explores adaptive sampling, it is unclear how such techniques apply to sound. Diffraction is an important effect in acoustic fields hard to predict with ray-based (i.e., geometric/Lagrangian) methods.

Referring to Fig. 1, the leftmost panel shows the reference field for a fixed player and varying emitter position. We plot the direct loudness parameter (L_{DS}) from parametric wave coding [14]. Dark areas indicate that a sound source located there is heard as attenuated at the fixed player location (red dot). We then generate a probe layout using our approach (right panel), and fix this probe budget for uniform sampling employed in parametric wave coding [14] (middle panel). In other words, we compare reconstruction quality to ground truth while holding cost (i.e., probe count) fixed. Note that only probes nearest the fixed player location participate in interpolation, yielding the visualized field determined by a weighted blend of the probes' emitter fields as defined in Section 3.

Fig. 1 shows the noticeable improvement in accuracy our method obtains compared to uniform sampling with visibility-based interpolation. The latter selects the only probe visible to the player (located in the central hall), leading to incorrect brightening of the central hall and darkening of the corridors near the player. Our technique ensures denser sampling in narrow areas at the expense of more open ones. More generally, uniform sampling aliases significantly, failing to sam-

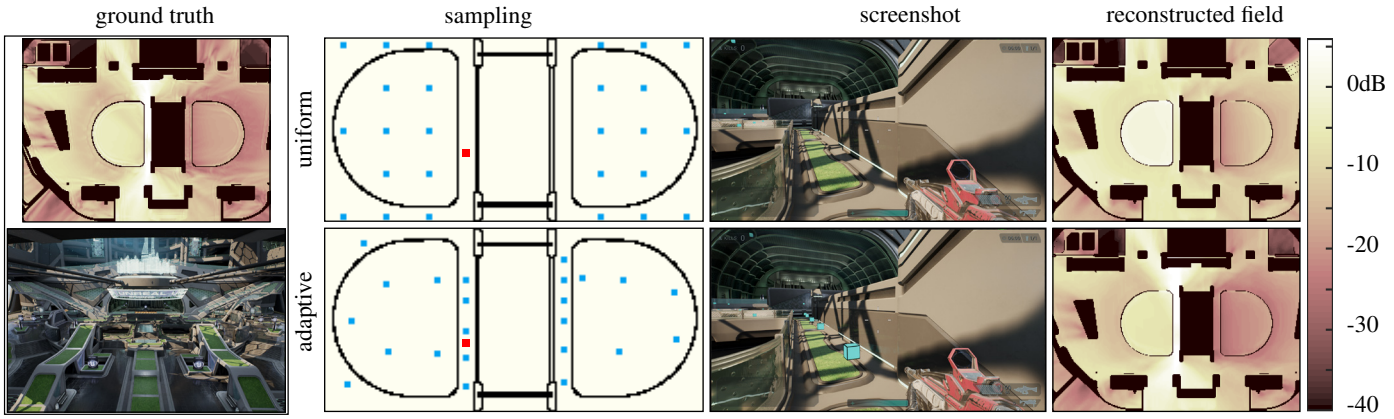

Fig. 9. Comparison of reconstructed fields of our method (bottom) against uniform sampling (top) on the HIGHRISE scene. Our method resolves the narrow walkway, resulting in reliable reconstruction of the precomputed field.

Fig. 10. Reconstruction accuracy for all parameter fields on EPICCITADEL.

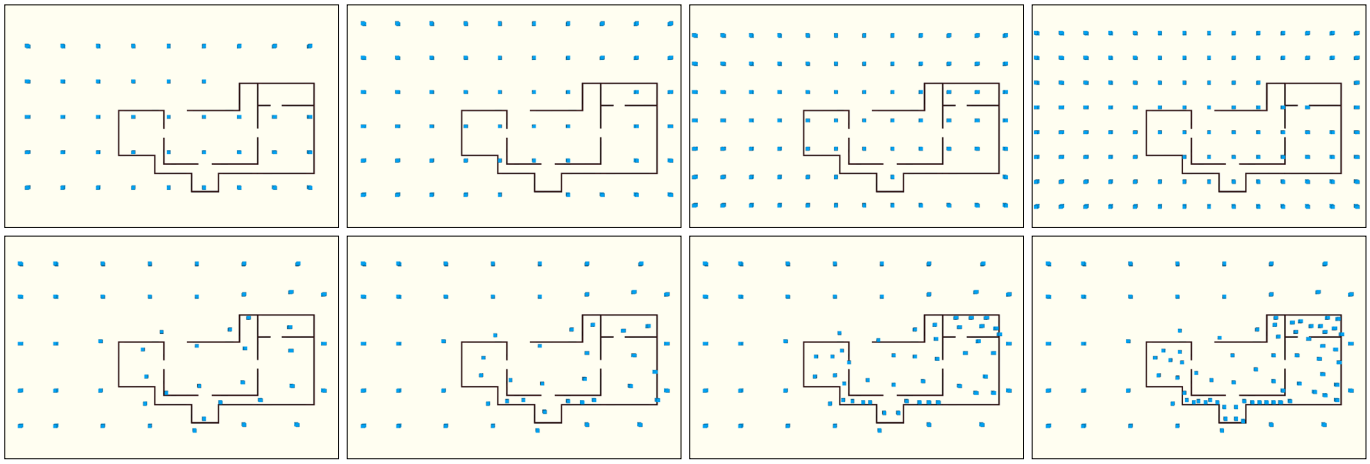

Fig. 11. Uniform (top) vs. adaptive (bottom) probe layout for equal probe budget. Left to right: sampling density increases as $s = \{1,2,3,4\}$.

ple the narrow horizontal corridors towards the bottom or the alcove near the top containing the green dot. Visibility-based interpolation exacerbates these sampling gaps to create dead spots. For example, a player located at the green dot finds no visible probes and thus nothing to interpolate, causing a decoding failure. Our method fixes this problem.

More complex game scenes in Fig. 8 and Fig. 9 yield similar improvement in accuracy, demonstrating the practicality of our technique. In Fig. 8, our technique samples reliably within the narrow vestibule. Uniform sampling omits sample "slots" that happen to be occupied by geometry and aliases inside narrow entrances and nooks. This results in audible loudness discontinuities, as illustrated in the accompanying

Fig. 12. Narrow tunnels and corridors are common in complex game scenes such as EPICCITADEL (left). Uniform sampling (middle) can miss such regions entirely. Runtime decoding will fail as the player enters such an area, causing unbounded rendering errors. Increasing global sampling resolution to about a meter resolves the issue but makes the precompute and memory cost impractical in such large game scenes. Our technique (right) adapts resolution locally to guarantee sampling within thin features avoiding such errors. Probes shown with blue boxes.

video. (Original results in parametric wave coding [14] avoid artifacts using a denser probe layout.) Uniform sampling selects the single visible probe near the entrance to the cathedral, overestimating loudness outdoors and underestimating it indoors, with errors as large as 20dB. Our method produces much better fidelity. Fig. 9 shows narrow walkways in an open scene. Uniform sampling misses the walkways and yields large errors while our method generates a string of probes down both walkways with greatly improved agreement to ground truth.

Fig. 10 shows all parameters used in parametric wave coding [14]: we obtain good agreement to ground truth for all four. Note that the direct sound field shows the strongest spatial variation, motivating our focus on it for experimental analysis. Note also that spatial variation in dark areas (e.g., for the reverb decay field) visually exaggerates the audible effect of such a highly attenuated result.

Most of our results use a sampling density of $s = 2$; Fig. 11 shows the effect of varying s, comparing to a uniformly sampled layout with the same probe budget in each case. As s increases, note how sampling density increases smoothly throughout the scene in the adaptive case, favoring narrow regions. Uniform sampling aliases first one region then another as density increases.

Fig. 12 shows probes (drawn as blue boxes here and in the video) available at the player position illustrated. With uniform sampling only a single probe appears at the end of a sloping tunnel; adaptive sampling provides probes along its whole length. See the supplemental video for walkthroughs including sound in all our demo scenes: HOUSE, EPICCITADEL, and HIGHRISE.

We apply a perceptual image metric, the structural similarity index metric or SSIM [26], to compare our results and previous work against ground truth. An overall similarity number is included in Figs. 1, 8, and 9. The number ranges in $[0,1]$ with a similarity of 1 indicating a perfect match. Similarity fields for each of our three scenes are shown in Fig. 13 which also reports overall improvement with our technique. While an image metric is not straightforwardly applicable to acoustic fields, these visualizations show perceptual parameter fields so that per-position audible features, such as loudness on logarithmic dB scale, map to per-pixel visible image differences; furthermore, temporal smoothness for a player moving through the scene corresponds to image smoothness over pixels. So we believe by comparing mean and variance over corresponding image patches, the SSIM metric provides some insight on acoustic similarity where perceptual metrics for spatially varying fields are not as fully developed. These quantitative results show our method improves fidelity compared to previous work. We also report RMS errors which show similar improvements.

Turning to our method's cost, we note that adaptive sampling represents a small fraction of the precompute cost. Tables 1 and 2 show precompute statistics for the scenes used in the paper. Time in both tables is measured in seconds. A single d evaluation is roughly a million times faster than a single probe simulation, while the total probe layout cost is roughly a thousand times faster than the total simulation cost over all the resulting pruned probes.

9 DISCUSSION

Our approach is motivated by the observation that the acoustic field is piecewise smooth with scene geometry causing arbitrarily large jumps. The acoustic rendering system must interpolate smoothly in open areas while still capturing these jumps across walls. The main error source is omitting probe samples in narrow enclosed places the player can visit. We thus give higher importance to areas of smaller diameter to ensure they get sampled. We don't claim our approach is optimal nor does it provide accuracy guarantees. Instead, we demonstrate good results in practice by adapting the sampling to the narrow regions, improving upon common issues related to state of the art uniform sampling with visibility-based interpolation.

Note that it is challenging to quickly and reliably predict spatial variation of global transport effects in all cases such as diffraction at portals, indirect scattering, soft shadowing from occluders, or caustics formed at the focal point of a concave reflector. This is true for light as well as sound propagation. The problem is made even harder because the location of the sound source is not fixed but rather tabulated over 3D space at precompute time. If error must be strictly bounded, there is little alternative than to compute acoustics for a dense sampling and then throw away unnecessary samples based on acoustic similarity criteria. As Table 1 shows, this would be tremendously time consuming and obviate any practical benefit of adaptive sampling. Our method of simply sampling more where the player can go is simple and fast (Table 2) and as we show, sufficient to adequately reconstruct piecewise smooth acoustic fields in complex and representative scenes, enabling smooth acoustic rendering in VR.

10 CONCLUSION

We present the first system to reliably capture precomputed acoustic effects for arbitrarily narrow scene features within a practical budget

Table 1. Time in seconds for single evaluation of local diameter, $d(x)$, vs. one probe simulation. Evaluating $d(x)$ is about a *million* times faster than a full acoustic simulation from the location. This makes it viable to evaluate $d(x)$ from the initial dense set of potential probe locations.

Scene	#Dense probes	All $d()$ evals	Single $d()$ eval	One probe simulation
House	10,465	25	0.002388	1,754
EpicCitadel	21,927	100	0.004560	2,120
HighRise	56,440	256	0.004535	2,077

Table 2. Time for probe layout vs. simulation of all pruned probes. Adaptive layout improves upon uniform sampling without substantially increasing precomputation time.

Scene	#Pruned probes	Probe layout	Simulation
House	53	291	92,962
EpicCitadel	381	521	807,720
HighRise	497	1,511	1,032,269

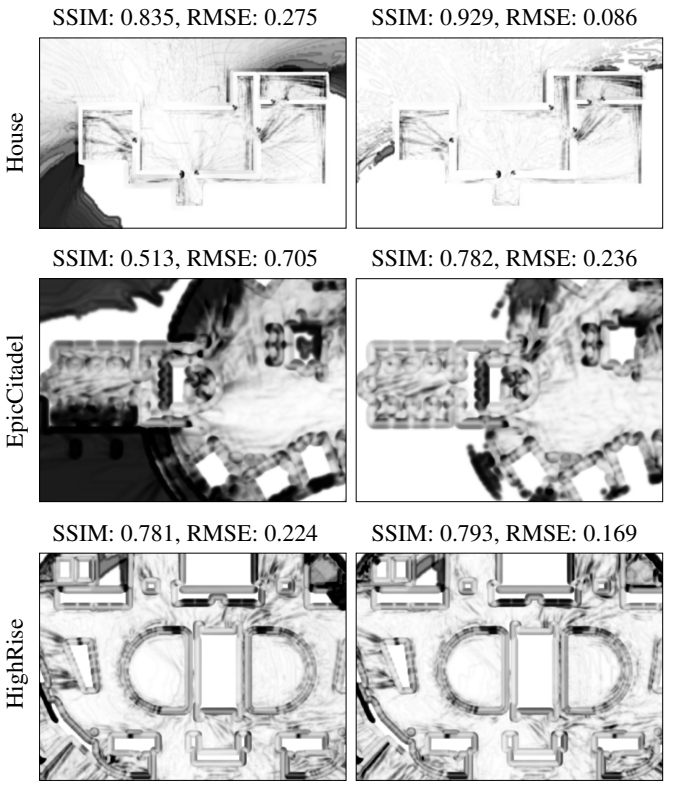

Fig. 13. Error statistics for parameter fields on our demo scenes: HOUSE, EPICCITADEL, and HIGHRISE. The left column shows the error result for the direct loudness field reconstructed using uniform probe sampling and visibility-based interpolation. The right column shows the result obtained by adaptive sampling and reachability-based interpolation. Lighter color represents larger SSIM values indicating closer match with ground truth. Lower root mean square error (RMSE) values are better.

for precomputation and runtime memory. The key idea is to smoothly adapt probe spacing to a novel, quickly-computable and stochastically-defined geometric measure of the local diameter around any scene point. We also introduce a new interpolation technique based on reachability that utilizes a simple model for diffracted transport, preserving the piecewise-smooth behavior of physical energy propagation in a scene while still reproducing jumps in the field at intervening walls. We observe significant improvement in accuracy over uniform sampling for the same probe budget on challenging cases with thin scene features frequently encountered in game scenes.

Our local diameter measure and adaptive sampling strategy is isotropic. Capturing directional dependence would allow sampling density dependent on direction, e.g., to sample more across a narrow corridor than along it. Investigating whether our ideas are useful for light probe placement also remains for future work.

REFERENCES

[1] Steam Audio Unity Plugin. `https://valvesoftware.github.io/steam-audio/doc/phonon_unity.html#placement-strategy`. [Online; accessed 27-July-2018].

[2] C. Cao, Z. Ren, C. Schissler, D. Manocha, and K. Zhou. Interactive Sound Propagation with Bidirectional Path Tracing. *ACM Transactions on Graphics*, 35(6):180:1–180:11, Nov. 2016. doi: 10.1145/2980179.2982431

[3] P. Debevec. Image-based Lighting. In *ACM SIGGRAPH 2005 Courses*, SIGGRAPH '05. ACM, New York, NY, USA, 2005. doi: 10.1145/1198555.1198709

[4] B. Hamilton and S. Bilbao. FDTD Methods for 3-D Room Acoustics Simulation With High-Order Accuracy in Space and Time. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 25(11):2112–2124, Nov. 2017. doi: 10.1109/TASLP.2017.2744799

[5] P. E. Hart, N. J. Nilsson, and B. Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, July 1968. doi: 10.1109/TSSC.1968.300136

[6] J. T. Hooker. Volumetric Global Illumination at Treyarch. In *Advances in Real-Time Rendering Course*, SIGGRAPH '16. ACM, New York, NY, USA, 2016.

[7] D. L. James, J. Barbič, and D. K. Pai. Precomputed Acoustic Transfer: Output-sensitive, Accurate Sound Generation for Geometrically Complex Vibration Sources. *ACM Transactions on Graphics*, 25(3):987–995, July 2006. doi: 10.1145/1141911.1141983

[8] H. Kuttruff. *Room Acoustics*. Taylor & Francis, 4 ed., 2000.

[9] M. McGuire, M. Mara, D. Nowrouzezahrai, and D. Luebke. Real-time Global Illumination Using Precomputed Light Field Probes. In *Proceedings of the 21st ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, I3D '17, pp. 2:1–2:11. ACM, New York, NY, USA, 2017. doi: 10.1145/3023368.3023378

[10] R. Mehra, A. Rungta, A. Golas, M. Lin, and D. Manocha. WAVE: Interactive Wave-based Sound Propagation for Virtual Environments. *IEEE Transactions on Visualization and Computer Graphics*, 21(4):434–442, Apr. 2015. doi: 10.1109/TVCG.2015.2391858

[11] Y. O'Donnell. Precomputed Global Illumination in Frostbite. In *Proceedings of the Game Developers Conference*, GDC '18, 2018.

[12] J. Pope, D. Creasey, and A. Chalmers. Realtime Room Acoustics Using Ambisonics. In *The Proceedings of the AES 16th International Conference on Spatial Sound Reproduction*, pp. 427–435. Audio Engineering Society, Apr. 1999.

[13] N. Raghuvanshi, R. Narain, and M. C. Lin. Efficient and Accurate Sound Propagation Using Adaptive Rectangular Decomposition. *IEEE Transactions on Visualization and Computer Graphics*, 15(5):789–801, 2009. doi: 10.1109/TVCG.2009.28

[14] N. Raghuvanshi and J. Snyder. Parametric Wave Field Coding for Precomputed Sound Propagation. *ACM Transactions on Graphics*, 33(4):38:1–38:11, July 2014. doi: 10.1145/2601097.2601184

[15] N. Raghuvanshi and J. Snyder. Parametric Directional Coding for Precomputed Sound Propagation. *ACM Transactions on Graphics*, 37(4):108:1–108:14, July 2018. doi: 10.1145/3197517.3201339

[16] N. Raghuvanshi, J. Snyder, R. Mehra, M. Lin, and N. Govindaraju. Precomputed Wave Simulation for Real-time Sound Propagation of Dynamic Sources in Complex Scenes. *ACM Transactions on Graphics*, 29(4):68:1–68:11, July 2010. doi: 10.1145/1778765.1778805

[17] N. Raghuvanshi and J. Tennant. Gears of War 4, Project Triton: Precomputed Environmental Wave Acoustics. In *Proceedings of the Game Developers Conference*, GDC '17, 2017.

[18] R. Ramamoorthi and P. Hanrahan. An Efficient Representation for Irradiance Environment Maps. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, pp. 497–500. ACM, New York, NY, USA, 2001. doi: 10.1145/383259.383317

[19] L. Savioja and U. P. Svensson. Overview of Geometrical Room Acoustic Modeling Techniques. *The Journal of the Acoustical Society of America*, 138(2):708–730, Aug. 2015. doi: 10.1121/1.4926438

[20] C. Schissler, R. Mehra, and D. Manocha. High-order Diffraction and Diffuse Reflections for Interactive Sound Propagation in Large Environments. *ACM Transactions on Graphics*, 33(4), July 2014. doi: 10.1145/2601097.2601216

[21] D. Schröder. *Physically Based Real-time Auralization of Interactive Virtual Environments*. Logos Verlag, Dec. 2011.

[22] A. Silvennoinen and J. Lehtinen. Real-time Global Illumination by Precomputed Local Reconstruction from Sparse Radiance Probes. *ACM Transactions on Graphics*, 36(6):230:1–230:13, Nov. 2017. doi: 10.1145/3130800.3130852

[23] P.-P. Sloan. Stupid Spherical Harmonics (SH) Tricks. In *Proceedings of the Game Developers Conference*, GDC '08, 2008.

[24] M. Taylor, N. Tsingos, and D. Manocha. Rendering Environmental Voice Reverberation for Large-scale Distributed Virtual Worlds. Technical Report 14-005, 2014.

[25] N. Tsingos. Pre-computing Geometry-based Reverberation Effects for Games. In *35th AES Conference on Audio for Games*, 2009.

[26] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, Apr. 2004. doi: 10.1109/TIP.2003.819861