Optimizing source placement for noise minimization using hybrid acoustic simulation

Nicolas Morales[a], Dinesh Manocha[a]

[a]*UNC Chapel Hill*

Abstract

Environmental noises can have a large impact on workplace productivity and health. We address the problem of designing large CAD structures that automatically meet the noise guidelines and standards. We present a novel approach that determines the optimal placement of sound sources such as generators, machinery, fans and HVAC appliances that can conform to these design requirements in indoor scenes. Our approach uses a hybrid sound propagation algorithm that combines elements of both geometric and wave-based techniques for handling higher and lower frequencies, respectively. We use an efficient simulated annealing approach for acoustic optimization that uses impulse response caching and sound source clustering to improve the convergence time. Finally, we present our approach in the context of a traditional CAD pipeline, where noise conformance requirements can be integrated during the early stages of the digital design phase. We highlight the approach's performance on many complex CAD models corresponding to an office and a warehouse.

1. Introduction

A leading problem in health and workplace safety is the effect of environmental noise on workers or building occupants. Environmental noise can have significant impacts on human health [6, 2], in addition to having an effect on workplace productivity and child learning ability. The industrial noise associated with manufacturing or production processes is regarded as a major occupational problem [42]. Additionally, noise level can have a negative impact on the animals [22]. In order to deal with these challenges, numerous regulations have been recommended to limit the environmental noise levels.

A key challenge is terms of computer-aided design of indoor structures like architectural models, factories, hospitals, or schools, is to ensure that they satisfy the noise and standards and regulations. Many engineering tasks revolve around optimizing an existing design so that it satisfies different criteria, including acoustic characteristics. In this paper, we address the problem of reducing the interior noise that is generated by different sources (washing machines, HVAC, fans, refrigerators, flush toilets, drains, etc.) in architectural models. The frequencies of these sound sources have a very large range: lower frequencies around $30-50$Hz and high frequencies that are more than $10KHz$. The noise levels are measured in terms of the sound pressure levels (SPL) generated from these sources at different points in the environment. Our goal is to automatically compute the location of these sources so that the resulting SPL is minimized throughout the environment or satisfies the noise standards.

The simplest methods that used for noise prediction enumerate different values of the characteristics of the absorbers or the enclosures in the scene or the source locations, and compute the maximum noise levels for each value. However, such trial and error methods can be very expensive and time consuming and may not be able to capture low frequency noise propagation effects, such as diffraction. Recently, various algorithms have been developed to automatically optimize acoustic characteristics of large models, including modifying shape [3, 26], material [42, 26], and topology [12]. Furthermore, these techniques can also be utilized to evaluate the noise characteristics. However, current methods either use simplified models based on acoustic diffusion for noise or do not provide sufficient accuracy in terms of modeling the low and high frequency components of noise. As a result, they do not provide sufficient accuracy for large architectural models.

A recent trend in designing large architectural or acoustic spaces is to use acoustic simulation or prediction methods that are sufficiently accurate as well as computationally fast. These simulation methods are frequently used to estimate the noise levels for different source locations or placements as part of overall optimization.There are generally two categories of acoustic propagation simulation: geometric and wave-based. Geometric techniques are based on ray-tracing, beam-tracing, and image source methods [1, 14, 48, 44]. These techniques work for higher frequencies and can handle a large number of sources in real time [40]. On the other hand, geometric techniques cannot accurately model low-frequency effects such as diffraction or scattering. These effects are often prevalent and noticeable at lower frequencies in terms of noise modeling. While some geometric techniques can represent lower-order scattering effects [13, 46, 11], psycho-acoustic studies have shown that geometric diffraction techniques

---

sound noticeably different from diffraction simulated using wave-based techniques [35]. Finally, state of the art geometric diffraction techniques can suffer in geometrically complex scenes [47]. In the second category of acoustic propagation simulation, the wave-based methods, that directly solve the acoustic wave equation using numeric methods. These include finite difference time domain (FDTD) techniques [5, 36], finite element methods [45], boundary element methods [9, 8, 38], pseudo-spectral techniques [16], and domain decomposition techniques [33]. In general, such wave-solvers can accurately compute the acoustic pressure field. However, their computational cost increases as a fourth power the simulation frequency and current wave-based methods are limited to lower frequencies, approximately less than 1 kHz or 2 kHz.

Main Results: We present a novel acoustic discrete optimization algorithm for source placement in large architectural or CAD models. Our primary contributions for this algorithm are as follows:

- An accurate and efficient hybrid sound propagation algorithm using a Linkwitz-Riley crossover filter for merging low and high frequency bands that can capture low-frequency wave effects such as diffraction while avoiding the cost of more expensive wave-based simulations.

- Source clustering of nearby sources to reduce the optimization search space between 2.5 and 8 times, which is necessary for efficient computations on scenes with a large degree of freedom for placement of sound sources.

- An efficient discrete optimization method for optimizing source placement that uses impulse response caching to improve convergence, reducing the effective algorithmic complexity of the heuristic optimization algorithm from worst-case $O(m!)$ to $O(m)$.

Our formulation performs discrete optimization to select from a finite number of possible positions for each position. The source positions are initially clustered to reduce the discrete search space. Then, in order to accurately compute the pressure field, we use a hybrid acoustic simulator that combines geometric and wave-based methods to take into account all source frequencies, including low and high frequencies. Our optimization models the environmental noise levels using an A-weighted curve model and computes the noise based on the impulse responses. Our hybrid acoustic optimization algorithm uses both geometric and wave-based techniques: geometric for the middle and high-frequencies of the simulation, and wave-based for the low-frequency bands. Thus, we gain the advantage of wave-based simulation at frequencies where diffraction effects are noticeable, but maintain computational efficiency of the algorithm overall by using geometric techniques for the higher frequencies. Additionally, we used a modified

simulated annealing algorithm that caches the impulse responses of source-listener configurations for efficient optimization convergence.

We highlight the performance of our method on different CAD benchmarks that mimic a variety of real-world workplaces like offices, warehouses, and industrial zones. These include industrial locations, where machinery can cause loud noise levels, and commercial office locations, where HVAC and light machinery noise can affect workplace productivity. To the best of our knowledge, this is the first algorithm that can optimize the source location in large CAD models to minimize the noise levels in selected regions.

The rest of the paper is organized as follows. We give an overview of prior work on noise modeling, hybrid sound propagation and acoustic optimization in Section 2. We give background on sound propagation algorithms and noise computations using impulse responses in Section 3. We present our novel optimization algorithm in Section 4 and highlight its performance on complex benchmarks in Section 5.

2. Prior Work

The problem of environmental noise reduction is a cross-disciplinary field with literature spanning across various fields from ecology [22] to urban planning [19] to computer-aided design and acoustics. Additionally, various regulations and recommendations are in place to control environmental noise including World Health Organization (WHO) recommendations [6] and other recommendations from regulatory agencies in various countries, including the Occupational Safety and Health Administration (OSHA) in the United States [31].

2.1. Environmental Noise Measurement

There is considerable literature on the measurement of environmental noise. This literature includes various works on environmental noise in certain geographic regions [18, 32, 52] and the assessment of the impact of noise reduction on production lines [50]. Ondet and Barbry predict noise in a room using ray-tracing simulation techniques [30]. Keränen et al [20] discuss the accuracy of geometric method for noise computation. Sequeira and Cortínez developed a method for optimizing the acoustic treatment of a structure using a simplified acoustic diffusion model [42]. Further literature includes discussions on environmental noise in urban environments [19, 7].

Various work has been done in the field of geometric sound propagation, including work that includes some scattering and diffraction effects. Some of these works include Tsingos et al. [46], Embrechts et al. [13], and the dissertation of Deines [11]. Various geometric room acoustic techniques are discussed in a survey by Savioja [39]. However, geometric techniques for diffraction and scattering cannot easily represent higher orders of diffraction

which can be prevalent at higher frequencies. Rungta et al. [35] show that participants in their study could observe a noticeable difference in propagated sound in up to three orders of diffraction when comparing a UTD-based geometric approach and a wave-based approach.

2.2. Hybrid Acoustic Simulation

In order to handle the lower and higher frequencies of noise, we use hybrid combinations of geometric and wave-based methods for simulation and optimization. These include techniques based on spectral decomposition that use wave-solvers for lower frequencies and geometric methods for higher frequencies [24, 43]. The computational complexity of these hybrid approaches is dominated by the wave-based methods that are performed over the entire acoustic domain, and current techniques are limited to small acoustic spaces. Different techniques have also been proposed for appropriate coupling at the interfaces between the geometric and numeric methods interfaces [51, 15, 49]. Our approach for handling interfaces is also based on these methods.

2.3. Acoustic Optimization

The problem of acoustic optimization has typically involved modification of a structure's shape, material, or topology. These works include Audioptimization by Monks et al. [26], geometry parameter modification by Bassuet [3], and optimal absorber placement [37]. While most acoustic optimization techniques use geometric techniques, some use wave-based methods [34, 28]. Generally the use of material optimization or shape optimization helps reduce the search space for the optimization algorithm; however inherently high degree of freedom optimization problems are explored in the field of topology optimization [12].

The problem of speaker and microphone placement is closely related to our goal. Speakers in this case are analogous to the location of noise-emitting machinery, while microphones correspond to the location of workers or patients in a hospital. As such, there have been various works in the problem of optimal speaker and microphone placement. Khalilian et al. [21] use methods for optimal speaker placement for sound field reproduction (SFR) problems. D'Antonio et al. developed an algorithm for optimizing speaker placement with constraints in addition to acoustic treatment and room dimensions for home theater systems [10]. Other techniques include using genetic optimization techniques for optimal loudspeaker and microphone placement [27].

3. Background

In this section, we give background on geometric and wave-based propagation methods, which are combined for hybrid acoustic simulation.

c	the constant speed of sound
t	time
$p(\vec{x}, t)$	pressure at location $\vec{x}$ at time t
p_0	reference sound pressure in Pa
ℓ_i	a listener position with index i
s_j	a source position with index j
S	audio of a sound source
P	sound pressure level in dB
L	the environmental noise level dBA
r	an impulse response

Table 1: Notation and symbols used in our acoustic solver and optimization algorithm.

Sound propagation is modelled by the time-domain acoustic wave equation:

$$\frac{\partial^2}{\partial t^2} p(\vec{x}, t) - c^2 \nabla^2 p(\vec{x}, t) = f(\vec{x}, t), \qquad (1)$$

where c is the speed of sound in a homogeneous media (typically $343\,\mathrm{m\,s^{-1}}$, the value in dry air at $20\,°\mathrm{C}$), $p(\vec{x}, t)$ is the pressure at location $\vec{x}$ and time t, and $f(\vec{x}, t)$ is a forcing function for the equation. The time-independent variant of this equation is also commonly used in wave-based techniques. Different methods exist for computing the sound propagation effects in a scene. In this section, we discuss the geometric technique we use for our algorithm along with the wave-based technique. Additionally, we use the impulse response generated by these approaches and show how to compute noise levels from the impulse response.

3.1. Geometric Sound Propagation

We use recent fast methods based on ray tracing for geometric sound propagation. These geometric methods assume that the wavelength of sound is much less than the size of features in the scene and then treat sound as rays, frustums, or beams. We make use of the ray-based sound propagation algorithm [41, 40] to compute paths by which sound can travel through the scene. Our approach uses path tracing with a cache of diffuse sound paths to reduce the number of rays required for an interactive simulation. We combine that with the image-source method and ray tracing for specular reflections. Our formulation begins by tracing a small number (e.g. $500 - 1000$) of rays uniformly in all directions from each noise source. These rays strike surfaces and are reflected recursively up to a specified maximum reflection depth (e.g. $40 - 100$). The reflected rays are computed using vector-based scattering, which uses a linear combination of the reflected rays with specular reflections and random Lambertian-distributed rays.

3.2. Adaptive Rectangular Decomposition

The adaptive rectangular decomposition technique [33, 25, 29] (ARD) is a time-domain wave-based solver that uses domain decomposition techniques to improve the efficiency of the solver. One of the advantages that a wave-based solver like ARD has over geometric techniques is the

(a) $t = 50$ (b) $t = 150$
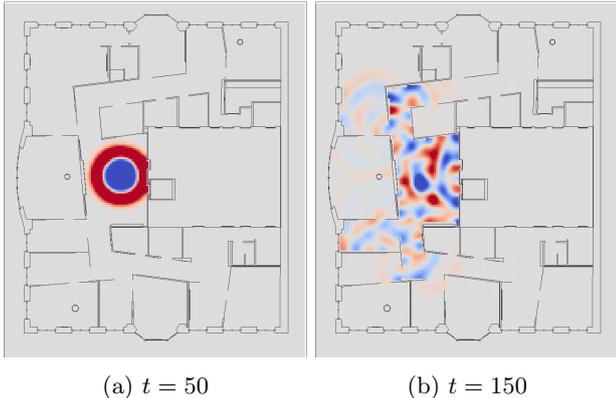
Figure 1: This figure shows the importance of diffraction (low frequency) effects in noise computation. It shows the pressure field at two different time steps $t = 50$ (a) and $t = 150$ (b) of are, red corresponds to positive pressure while blue is negative pressure. In this slice of an office environment, a sound source placed near the elevator hallway propagates through the various office and work areas. Diffraction effects such as this are challenging to compute using geometric solvers but are inherent to wave-based solvers. Prior methods for noise modeling do not take such effects into account.

Figure 2: The A, B, and C-weighting curves. These are used to account for the loudness of a sound perceived by a human being. A-weighting is commonly used for noise measurements although C-weighting is often desirable because of its flat curve. In our noise modeling results, we used the A-weighting, though it is possible to use other curves as well.

ease and accuracy of computing diffraction effects (Figure 1). ARD is also considered a low-dispersion solver, in that it allows a much coarser spatial discretization size compared to similar methods such as FDTD. The grid cell size of the discretization in ARD is 2 samples per wave length (and thus inversely proportional to the frequency).

The ARD algorithm works by dividing the solver domain into three-dimensional cuboidal regions. In each individual sub-domain, the acoustic wave equation has a known analytical solution. That means that within a particular sub-domain, we can determine the exact solution to the wave equation given the assumption of perfectly reflective boundaries.

At the interfaces between sub-domains, the ARD method uses FDTD-like stencils for propagated waves across the boundaries. This allows for more complicated boundaries, including partially absorptive boundaries (such as walls) or pass-through boundaries between two sub-domains that do not have a wall between them. Acoustic materials are modeled by adjusting these stencils.

3.3. Impulse Response and Noise Calculation

We use the concept of impulse responses for noise calculation and optimization. The impulse response (IR) is a function that measures the way in which sound pressure waves reach a listener position after being excited by an impulse sound similar to a Dirac delta function. It is defined for a fixed source-listener pair in the environment. As the source moves, the IR is updated. Impulse responses are important in any kind of acoustic metric computation because they represent the way in which sound propagates between the source-listener pair. In some ways, the IR can be considered the acoustic signature of a room. In ARD, the source listener is a Gaussian derivative centered around

the source frequency in the spectral domain. In geometric sound propagation, this takes the form of a Dirac delta function.

3.3.1. Convolving IRs with Noise Sources

The IR gives us a function characterizing the way in which sound is propagated throughout an environment. We are specifically interested in the sound levels propagated by machine noises, which can have variation from operation variability or from impulse sounds such as a hammer impact. These sound clips can be recorded using a microphone and then re-scaled to the estimated peak noise level. In order to determine how the sound clip is propagated throughout our environment, we use the impulse response. By convolving the sound clip with the impulse response, we yield the propagated sound:

$$P = 20 \log_{10} \frac{r * S}{p_0} \, \text{dB}, \qquad (2)$$

where r is the possibly re-sampled pressure impulse response, S is the recorded sound clip in pascals, and p_0 is the reference pressure at the limit of human hearing at 2×10^{-5} Pa. We use this convolution process on the impulse response to compute how various sound clips such as machinery noise or appliance noise propagate throughout the environment. In noise computation, the convolved sound levels are important because their frequency data reflects both that of the impulse response and that of the sound clip. As we will discuss in the next section, frequency data is important in noise computation.

3.3.2. Computing Noise Exposure

There are a variety of methods for computing environmental noise, but most regulatory agencies use weighted decibel levels (known as A-weighting, B-weighting, or C-weighting) and time-weighted average (TWA) metrics [31].

Weighted sound level curves are designed to account for the loudness perceived by human hearing. There are usually three curves: A, B, and C. A-weighting is the most commonly used curve by equipment manufacturers and regulatory agencies, while C-weighting is often used when a flat frequency response is desired [4]. B-weighting is mostly useful for higher sound pressure levels, but is rarely used nowadays [31]. Figure 2 shows the gain of each of these curves across the range of human hearing. In our work, we focus on the A-weighting curve. It is the most popular and is also standard in many regulations.

To determine the averaged A-weighted sound level from recorded or simulated sound, we compute the average sound level in each frequency band f. The formula for the average A-weighted sound is as follows:

$$L = \log_{10} \sum_f 10^{\frac{P(f) + A(f)}{10}} \, \text{dB}, \qquad (3)$$

where $P(f)$ is the sound pressure level in decibels of the frequency band corresponding to f and $A(f)$ is the A-weighting adjustment at frequency f. The A-weighting adjustment is given by [17]:

$$
\begin{aligned}
U(f) &= (f^2 + 20.6^2)(f^2 + 12194^2), \\
V(f) &= (f^2 + 107.7^2)(f^2 + 737.9^2), \\
R(f) &= \frac{12194^2 f^4}{U(f)\sqrt{V(f)}}, \\
A(f) &= 20 \log_{10} R(f) + 2.0
\end{aligned}
\qquad (4)
$$

In frequency space, the final adjusted environmental noise level per frequency band is:

$$L(f) = rS + A \, \text{dB}, \qquad (5)$$

where A is an adjustment on the final convolved sound pressure level computed in section 3.3.1.

4. Sound Source Optimization

In this section, we describe the overall source placement algorithm for noise optimization. In terms of optimization, we are interested in the minimization of noise at a set of n listener positions $\ell_1 \ldots \ell_n$. These listener positions are located in the areas where noise can be a problem, such as a work location or a hospital bed. The noise in the listener positions are induced by a set of m sound sources $s_1 \ldots s_m$. In our optimization formulation, we are interested in the placement of the sources according to constraints such that the noise reaching the listener positions is minimized. Given this goal, we can derive the following objective function for our optimization process:

$$\arg\min \left(\max_i \sum_j L\left(\ell_i, s_j\right) \right), \qquad (6)$$

where $L(\ell_i, s_j)$ is the averaged A-weighted level across all frequency bands induced by a sound source s_j on the listener position ℓ_i. We sum each of these A-weighted levels to get the total noise for all frequency bands at listener ℓ_i, generated by all the sound sources. We are particularly interested in the listener position with the maximum noise level. While the maximum operation could be replaced by an average or other form of statistical analysis, we are interested in this formulation of the problem that ensures that that the noise at any location in the environment is not too high, and below the guideline. This is useful in a situation in which an acoustic designer must conform to a set of regulations in which no part of the work environment can exceed safe noise levels.

One major issue is that the evaluation of L is non-trivial and there is no easy closed-form solution. This is because it involves evaluating both the geometric and ARD solver one or more times. The geometric solver is used to compute the response at the higher frequencies and the wave-solver is used to compute the response at the lower frequencies. Therefore, in our algorithm, we explore ways to efficiently compute the result. Typically, the complexity of the propagation algorithms is a linear function in the number of sources. In order to perform efficient computations, we use three techniques:

- We use sound source clustering to reduce the optimization search space.

- We use efficient simulated annealing approach for optimization that basically performs faster search for a solution.

- We introduce impulse response caching to accelerate iteration time for the optimization.

The combination of these three techniques can considerably improve the runtime performance and makes it possible to optimize the source positions in large CAD scenes for noise minimization. Figure 3 summarizes these elements in the context of our overall algorithm. Next, we describe each of these techniques in detail.

4.1. Hybrid Propagation for Noise Computation

In order to compute the noise metrics for a specific source location, we use a novel hybrid sound propagation algorithm. The noise is computed directly from the source sound convolved with the set of impulse responses corresponding to the combinations of available listener locations and source locations.

Using a geometric solver for this computation is fast, but it does not easily capture low frequency diffraction effects. On the other hand, wave-based solvers are expensive for higher frequencies. Moreover, we need to compute the field response for every single source that contributes to the noise level in the environment. This means that during every iteration of our algorithm, we would need to compute the pressure field multiple times. Since wave-based
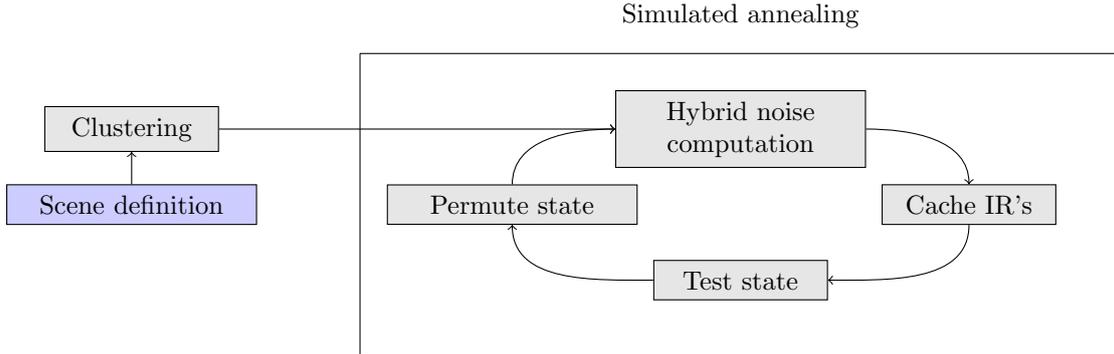
Figure 3: **Our optimization algorithm:** The input to the algorithm is the scene representation (i.e. the triangulated CAD model), which includes possible source locations and the listener regions. The state corresponds to the current set of source positions. We permute between them to move to the next set of positions. We perform source cluster before simulated annealing algorithm. The various components of simulated annealing including noise computation, IR caching, testing states for acceptance based on metrics, and permuting the states to get the neighbor states. Section 3.3.2 covers the details of how noise is computed.

methods scale with $O\left(f^4\right)$, where f is the highest simulation frequency, this process becomes even more expensive for high-frequency noise.

On the other hand, we can take advantage of geometric sound propagation for higher frequencies where diffraction effects are less prevalent. In large models, ARD is used for all frequencies up to $500\,\text{Hz}$ (or $250\,\text{Hz}$ for large models where ARD is more expensive), where its computational overhead is reasonable. After computing the results using wave-based and geometric methods for separate frequency ranges, we need to calibrate them.

Our algorithm uses a frequency-domain separation as opposed to a spatial-domain separation where a more accurate wave solver is used for detailed or complex objects [51, 15, 49]. Similarly to other frequency-domain separation techniques [24, 43], we use a wave-based solver for lower frequencies and geometric for higher frequencies. However, for lower frequencies we use the ARD solver rather than an FDTD solver. This allows us to obtain some performance improvements because the cell size for ARD is much coarser than that of the FDTD solver.

To combine the methods, we low-pass the wave-based impulse response and high-pass the geometric response using appropriate filters. In order to avoid the ringing artifacts, we use a Linkwitz-Riley crossover filter [23]. These use cascading Butterworth filters that help avoid ringing artifacts at the crossover locations. Figure 4 shows the details of this computation.

The calibration of the two approaches, based on wave-based and geometric methods, depends on amplitude normalization of the two IRs. We use the direct sound for both these methods for normalization; this corresponds to the part of the sound that reaches a listener directly without reflecting, scattering, or diffraction. Under these conditions, the sound pressure level at a distance d from a sound source is proportional to $\frac{1}{d}$. Given a sound source of power w in watts, our sound pressure at distance d is computed as:

$$p = \sqrt{\frac{w}{4\pi d^2}}\, z_0, \tag{7}$$

where z_0 is the atmospheric impedance value (generally $413\,\text{N}\,\text{s}\,\text{m}^{-3}$). If we pick $w = 1$ for our sound source power in the impulse response, we can compute an impulse response independent of the sound source clip S. While convolving the IR with S later, as long as S has an appropriate scaling for its sound power, we get the propagated sound at the correct loudness values.

4.2. Source configuration

The possible locations of sound sources in our algorithm can be defined by the configuration space C. This is a subset of $\mathbb{R}^3$ where user-defined constraints and collision constraints are enforced on the sound source locations (e.g. the sound source cannot be in a wall). The goal of the optimization algorithm is to find the source positions s_i within the configuration space where the objective function (equation 6) is minimized. However, the user and collision constraints must be adhered to during the optimization process.

Initially, we only enforce the user constraints. This reflects the natural constraints of placing the machinery or the equipment, such as fixtures or other structural or utilitarian constraints. For example, in a home, a washing machine can only be placed on a specific washing machine fixture where it is connected to the water supply. Similarly, there are fixed wiring locations for the fans or HVAC. In an industrial context, a generator may need to be placed near machinery operated by a worker. The user specifies these constraints using a plugin to the popular open source modelling tool, Blender.

The space defined by these initial constraints is discretized into a set of points S. These points $s_i \ldots s_m$ define the possible locations where a source can be placed and are determined by a stratified sampling of the configuration space C. The sampling density ρ is empirical in our algorithm, but should be chosen densely enough such that features of the configuration space are well sampled.

6
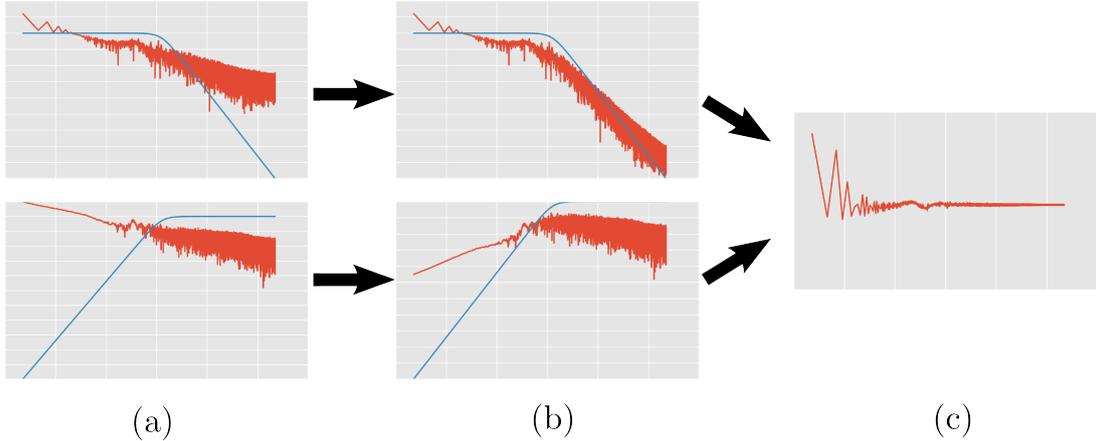
Figure 4: **Combining impulse responses using hybrid methods:** (a) The top impulse responses is computed using ARD and the bottom impulse response is computed using a geometric methods. These impulse responses are then filtered using the Linkwitz-Riley crossover filter [23] shown in (b). Finally the filtered IRs are added together to produce (c), which represents the accurate IR over the entire band of the noise source.

(a) Original configuration   (b) After clustering
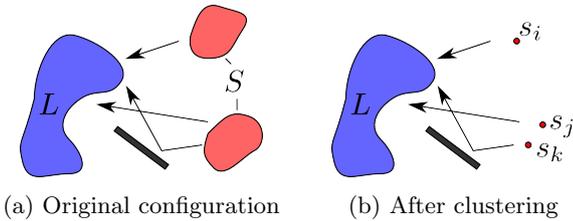
Figure 5: Clustering allows us to represent large areas of possible source locations by a single (virtual) source location. Our clustering approach is guided by the fact that the noise difference between the representative source and the other sources in the cluster is less than a threshold. In our experiments, we used a threshold of 1 dB, or approximately the just-noticeable difference (JND) for the human auditory system. JND is governed by psycho-acoustic studies.

Collision constraints are then enforced next. We use the low-frequency spatial discretization of the ARD method to remove possible source positions that are within wall cells. This ensures consistency between the wave-based and geometric methods. Wall cells are determined by a flood-fill algorithm once the CAD model geometry is voxelized in the ARD pre-processing. An "air" voxel is marked by the user and any voxel not reachable from that point is considered to be a wall.

4.3. Sound Source Clustering

One consequence of this is that the size of the search space for the optimization problem can very large. This can yield many iterations of the optimizer, and each iteration requires a full evaluation of both geometric and ARD per source-listener pair. This is very slow as a result of using the more accurate wave-based solver, which can take a few minutes to solve a single instance of the source-listener problem. Therefore, we find that the running time of our algorithm is heavily dependent on the total volume of constraint regions.

One way of ameliorating this problem is by exploiting the property that some neighboring samples inside these constraint regions have essentially the same A-weighted level in terms of the noise characteristics. This can be evaluated using a free-field region of space, i.e. one with no obstacles. Two sound sources of equivalent distance and acoustic characteristics, barring interference, will result in the same sound pressure level. This criteria is explained in Figure 5, where source regions are replaced by representative sources according to the clustering algorithm.

Geometrically, it is difficult to determine which portions of the model actually correspond closely to this free-field condition from a given CAD model. This is a consequence of the complexity of interactions between the sound waves, surfaces, and the obstacles. However, we can use sound propagation algorithms to determine which sample locations have similar sound levels. Using geometric sound propagation for this computation is appealing for two reasons. First, that geometric sound propagation is orders of magnitude faster than wave-based techniques and the current methods can handle large scenes in tens of millisecond. Secondly, it provides an accurate sound pressure level for higher frequencies. Although low-frequency wave phenomena are important, metrics such as A-weighting weight lower frequencies less.

Therefore, we evaluate the sound pressure level from each sample point and implement hierarchical clustering for the sample points. The threshold for hierarchical clustering is set to be a loudness value that is an acceptable error in the optimization process. In other words, for two sources s_j and s_k, they can be clustered together if they are near each other and

$$\left\| \max_i L\left(\ell_i, s_j\right) - \max_i L\left(\ell_i, s_k\right) \right\| < \tau, \qquad (8)$$

where the threshold τ is generally the just-noticeable difference (JND) for human hearing. This clustering allows

7

us to select representative samples for each constraint region. In the case where the sound pressure level for multiple sound sources in the constraint region changes greatly, we use a separate representative sound source for each cluster.

As a result of this clustering, we significantly reduce the search space for our optimization algorithm. The amount depends on the location of sources, but we experienced a reduction on some of our benchmarks by about 5x to 7x for a clustering threshold of $1\,\mathrm{dB}$ (approximately the just-noticeable difference of human hearing). This reduction improves the convergence rate and performance of our algorithm.

4.4. Simulated Annealing

Input : Listener regions Ω
    Initial temperature T_0
    Cooling rate α
Output: Optimal source locations q
initialize simulator;
$\vec{s} \leftarrow$ ComputeInitialState()
$q \leftarrow \sum L_{max}(\Omega, s_i))$;
$T \leftarrow T_0$;
while $T > 1$ **do**
   $\vec{s} \leftarrow$ PermuteState($\vec{s}$) /* Compute new state */
   $q' \leftarrow \max_i \sum_j L(\ell_i, s_j)$;
   if TestState(q,q',T) **then**
     | $q \leftarrow q'$;
   end
   $q \leftarrow$ Min(q,bestState);
   $T \leftarrow \alpha T$;
end
Procedure TestState(q,q',T)
   if $q' < q$ **then**
     | **return** true;
   end
   $p \leftarrow \exp \frac{q-q'}{T}$;
   return $p <$ Rand(0,1);

Algorithm 1: The simulated annealing algorithm for discrete acoustic material optimization. Each iteration of the algorithm involves computing the acoustic pressure field using both the geometric acoustic solver and ARD.

After sound source clustering, we end up with a set of discrete locations that a sound source s_j can be placed. This is fundamentally a combinatorial problem and implies the need for a discrete optimization approach. The objective function in Equation 6 can therefore be minimized by using a simulated annealing approach. This technique is efficient for large search spaces — even after clustering, there can often be on the order of millions of total combinations of source locations. Brute force solutions for search spaces of this size are completely intractable. Even at an optimistic minute per iteration for a naive implementation an exhaustive search would take years.

The advantage of simulated annealing is that it can avoid local minimum in the search for a global minimum. It minimizes an objective function by randomly permuting the state, accepting new states that have lower energy. Additionally, in order to avoid local minimum, the annealing algorithm will sometimes also accept less optimal states. The probability of this happening depends on the system temperature, T. Each iteration, this temperature decreases by a factor of the cooling rate, α. As the system cools, the algorithm is less likely to select less-optimal states.

4.4.1. Simulated Annealing State

The state variable $\vec{s}$ represents a list of all the sound sources in the scene. The end goal of our optimization algorithm is to determine a state $\vec{s}$ such that the maximum noise at all listener positions is minimized. Algorithm 1 shows how the state is iterated on to yield a global minimum. First, the state is assigned through a random shuffle (in ComputeInitialState). The state variable gets permuting every iteration by PermuteState. The energy q' of this new state is then used to determine whether the new state is accepted or rejected. In our algorithm q is simply the maximum noise value across all listener positions, or $\max_i \sum_j L(\ell_i, s_j)$. In simulated annealing a state q' that is better than the old state q is always accepted. If the converse is true, the new state may still be accepted depending on the temperature T.

4.5. Impulse Response Caching

Simulated annealing provides an efficient heuristic for minimizing in a discrete search space, even if the search space is combinatorial in nature. However, even for a reduced number of iterations computing the impulse responses using the ARD wave-based solver is expensive.

For each simulated annealing iteration, our algorithm must compute the impulse response between each source and listener pair in the state. It's worth considering that for n listeners and m sources, the total number of these pairs is $O(nm)$, much less than the size of the total search space which is $O(m!)$.

As a direct consequence, we can cache source-listener pairs that already have a computed IR. Instead of redundantly computing an already computed IR, we retrieve it from the cache. This significantly reduces the cost of later iterations of the simulated annealing algorithm, particularly since the wave-based solver would otherwise have to be executed once for each non-cached pair that is in the current iteration state.

Even though the search space is still $O(m!)$, exploring all source-listener pairs means that no more impulse responses need to be computed. Thus, our algorithm can be stated as essentially having $O(nm)$ asymptotic time given that the iteration time for cached impulse responses approaches 0. However, in practice, since the ARD method is

significantly more expensive than the geometric technique and dominates our computation time, the asymptotic time can be further reduced to $O(m)$. This is because ARD computes a global solution to the wave equation, so the running time is not dependent on the number of listener positions.

5. Results

We have evaluated our algorithm on three different and complex CAD environments: an office, a warehouse, and an industrial zone. These scenes were obtained from existing model databases such as TurboSquid that include some architectural models and were adjusted so that the scaling and size were appropriate and would be similar to CAD models used in designs for noise analysis. In addition, for each model, we determine a set of listener locations and the possible source locations.

Along with the CAD models, we used a variety of sound clips to represent the variety of sound or noise sources that might occur in an industrial or workplace environment. The spectrograms of these sources are shown in Figure 7. Table 2 shows a summary of the scenes and the sound sources used in these models. Additionally, the table shows the advantage of our clustering algorithm: the clustering ratio for each scene represents how much the number of source locations was reduced by. For example, the Warehouse scene initially had 25 possible source locations, with a sampling distance of 0.7 meters, but after clustering had a 2.5 times reduction to yield 10 source locations. This considerably improves the performance of sound simulation and sound propagation algorithms.

5.1. Performance

One key benefit of our algorithm is the efficiency in optimization using clustering and IR caching. This dramatically improves the convergence and runtime perform of our acoustic optimization algorithm. In Figure 8, we demonstrate how the IR caching algorithm improves our iteration time. Without caching, the iteration time would be roughly constant. Instead, we observe that as the number of iterations increase, we are more likely to have covered all of the $O(mn)$ combinations of source and listener pairs.

Additionally, Table 3 shows how our algorithm behaves on different types of benchmarks. We compute the total running time of the algorithm over all iterations, e.g. we use 100 iterations in our simulated annealing process. The primary factor in performance is the number of grid cells. This corresponds to the number of voxels that ARD uses for its regular grid when solving the wave equation. On some scenes, such as office, we had to artificially use a higher number of grid cells than required by the stability conditions of ARD [33] to generate highly accurate results.

On the other hand, the number of triangles in the scene has little effect on the wave-based algorithm since operates on a grid, not a triangle mesh and the grid size is governed by the maximum simulation frequency. The complexity of the mesh does affect the performance of the geometric solver, but this is negligible compared to the time required for the ARD solver. The geometric solver is based on ray tracing, that uses a bounding volume hierarchy to accelerate the computation. Its complexity is logarithmic in the number of triangles.

5.2. Noise Minimization

Using our algorithm, we can pick the best source locations that will minimize the noise for a set of listener locations. Figures 9, 10, and 11 show the full-field noise levels for the entire domain as a result of our optimization process. In the figures, we show both the regions that contain different listener positions in addition to the source location regions. Our algorithm gets quite close to optimal noise levels in these regions.

Additionally, we show result on how the noise field has changed. Figure 12, 13 and 14 show how optimization changed the noise distribution in the scene.

5.3. Error Analysis

In addition to performance and noise minimization results, we analyzed some of the error properties of our hybrid simulation approach. We chose the office scene for the first analysis because of its complex geometry and the prevalence of diffraction effects where geometric approaches struggle. Figure 15a shows our experimental setup. Additionally, we conducted an experiment on the warehouse scene, where the presence of many shelves makes diffraction effects important. The setup for this scene is shown in Figure 16a.

Figure 15b and Figure 16b show the error of the frequency responses of the hybrid and geometric approaches induced by a Gaussian derivative source with a primary frequency of 125 Hz. The hybrid approach in this case used a critical frequency of 500 Hz to transition between wave-based and geometric components. In the office scene, in the octave frequency band centered around the source, the hybrid approach had an average error of only 6 Hz, while the geometric approach had a much more significant error of 27 dB where it did not compute the diffractive properties of the propagated sound. Similarly, in the warehouse scene, the hybrid approach had an average error of 8 dB on the frequency band, while the geometric approach had a higher error of 36 dB error.

6. Conclusion, Limitations and Future work

In this paper, we described an efficient discrete optimization technique using hybrid acoustic simulation that minimizes acoustic noise at specific listener areas. Our goal is to optimize the source locations. Our approach takes advantage of source clustering and impulse response caching in order to reduce the total search space of the algorithm in addition to reducing the iteration cost of the algorithm.

(a) The office scene	(b) The warehouse scene	(c) The industrial scene
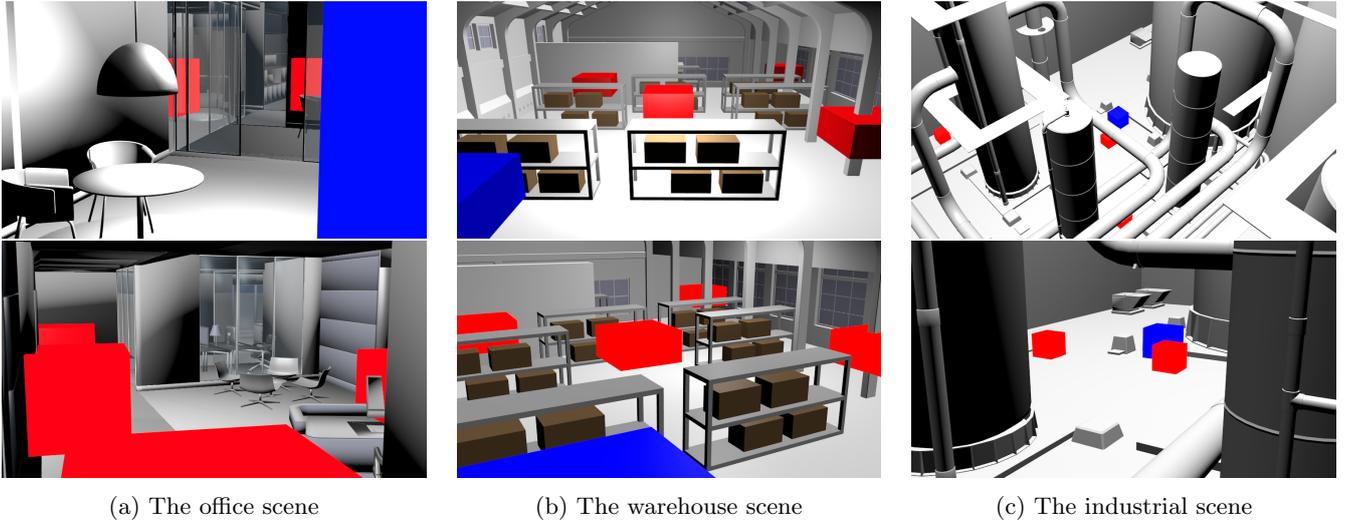
Figure 6: We have evaluated our algorithm on three complex CAD benchmarks: the office scene, the warehouse scene, and the industrial scene. All three benchmarks represent different environments in which one would want to minimize noise levels. In this figure, the blue rectangular regions represent our listener regions; we sample from this region in order to get source-listener pairs. Additionally, the red regions represent possible sound source locations. For example, in the office scene some of the source locations represent areas in which an air conditioning duct could be placed. We also sample from these locations, clustering the samples in order to reduce the total number of possible source locations to a representative set.

Scene	Sound Sources	Noise level	Avg. source region size	Sampling density	Num. source regions	Clustering ratio
Office	1x machine (70 dB), 5x HVAC (50 dB)	38.48 dB	$1 \times 1 \times 1$ m	0.4	11	7.55x
Warehouse	3x heavy machine (85 dB)	59.79 dB	$2 \times 2 \times 1$ m	0.7	25	2.5x
Industrial	5x heavy machine (85 dB), 2x generator (90 dB)	50.14 dB	$1.5 \times 1.5 \times 1.5$ m	0.7	104	8.0x

Table 2: Table: We highlight the benchmark scenes, their sound sources, the minimized noise level, and the clustering ratio. In some cases we used multiple types of the same source. For example, in the Industrial scene we optimized the placement of two generators. The clustering ratio is used to evaluate the performance of our clustering algorithm. The higher value of the clustering ratio indicates greater reduction in possible source positions. For example, a clustering ratio of 5.36X implies that number of sources was reduced to approximately $\frac{1}{5}$ of their original number.

Scene	Crossover Frequency	Number Tris	Number cells	Total time
Office	500 Hz	973373	2.5m	1398.02 s
Warehouse	500 Hz	21188	3.7m	1254.84 s
Industrial	250 Hz	51802	14.3m	4232.0 s

Table 3: The total running time of each scene along with the number of cells used for the ARD algorithm and the number of triangles in the CAD models corresponding to our benchmarks. Additionally, the crossover frequency for the hybrid simulation is listed. The most important factor int he running time is the number of cells, since the ARD wave-solver is the main bottleneck in the algorithm. The geometric sound propagation based on ray tracing only takes a few tens of milliseconds performance. Due to the use of IR caching and source clustering, we considerably reduce the runtime performance.

(a) HVAC

(b) Machine

(c) Heavy machine

(d) Generator
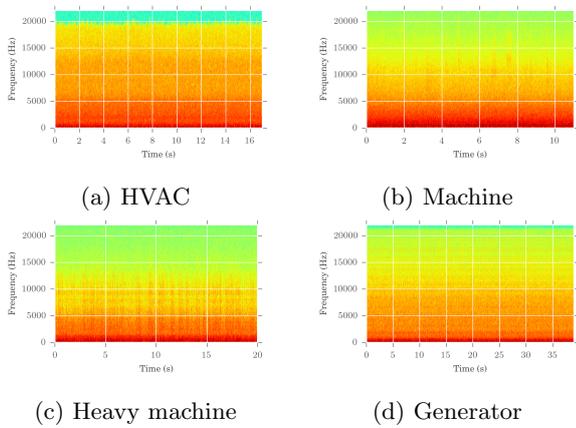
Figure 7: The spectrograms of various sound sources that we used in our benchmarks and evaluated their noise effects. The red color represents higher amplitude in those frequencies at that point in time. Many of these sound sources have a wide range of frequencies; for example the HVAC sound source has contains very high and very low frequencies. We are able to optimize across this large frequency range because our hybrid sound propagation algorithm can easily deal with both low-frequency and high-frequency effects.

Figure 9: Noise field slice for the office scene calculated with ARD using the impulse response at every location. The noise field is minimized in our listener region (blue). The red regions are areas where sources can be placed. Each red area can have multiple source locations depending on the clustering.

Figure 8: The time spent during each iteration for the three benchmarks. Importantly, as the number of iterations increase, it is more more likely that all of the listener-source pairs corresponding to the current state are already cached. This helps improve the running time of the simulated annealing algorithm even if the number of iterations is large. In this figure, iterations after step 10 do not need to calculate new impulse responses and just use the cached values.

Figure 10: Noise field slice for the warehouse scene calculated with ARD using the impulse response at every location. The noise field is an output of our algorithm, where the goal is to minimize the listener region (blue). The maximum noise level in the blue region is used for computing the energy for the simulated annealing process.
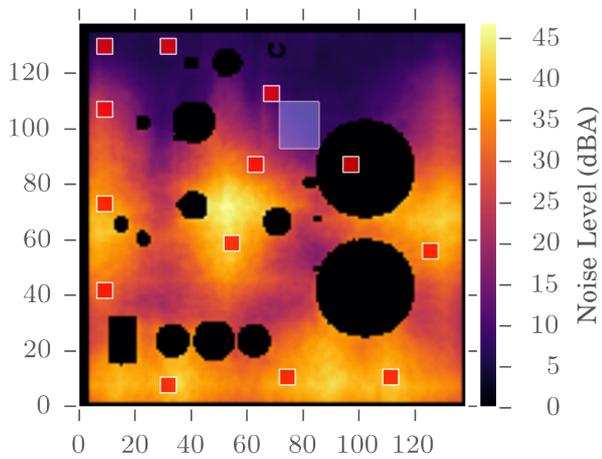
Figure 11: Noise field slice for the industrial scene calculated with ARD using the impulse response at every location. The minimization region, where the listener positions are, is shown in blue, while the source constraints are shown with red markers.
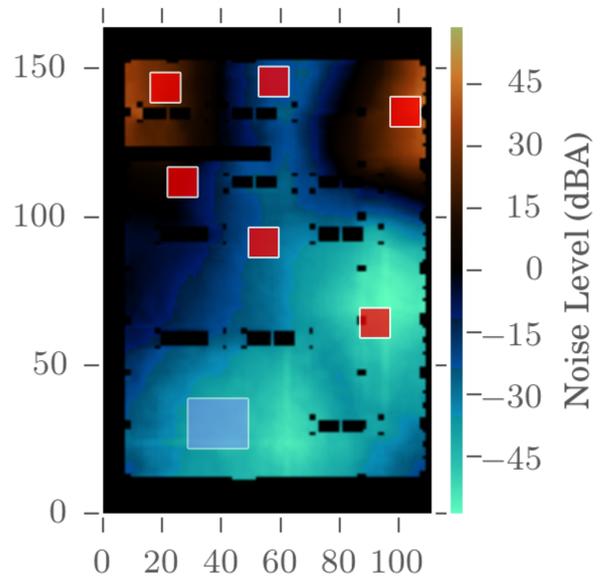
Figure 13: Change in the noise field for the warehouse scene. After optimizing, the noise in the blue regions is minimized. Note that the listener locations are in the blue regions.
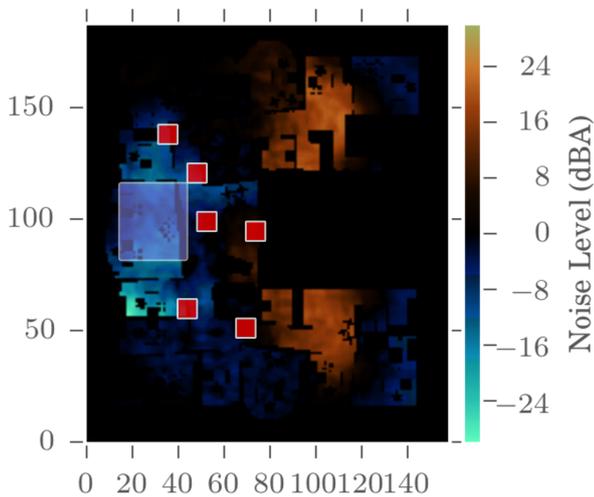
Figure 12: Change in the noise field for the office scene. Blue areas are regions in which the noise was reduced and orange areas are regions in which the noise was increased as a result of the changes. Note how the listener positions (the blue box) are located in the blue regions.
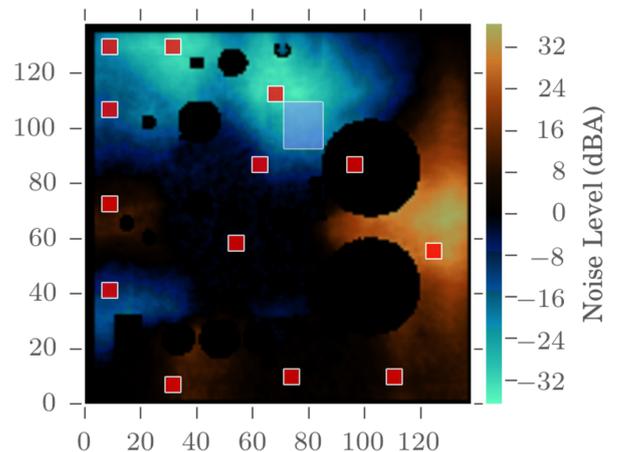
Figure 14: Change in the noise field for the industrial scene. Blue areas are regions in which the noise was reduced and orange areas are regions in which the noise was increased as a result of the changes. As a result of our algorithm, the noise level in the listener regions (the blue box) was reduced.
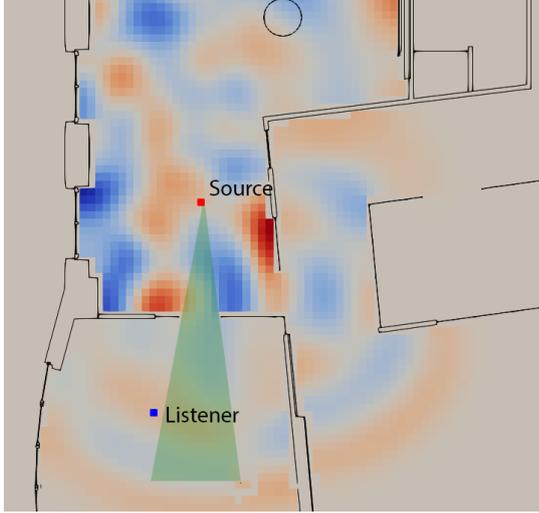
(a) Experimental Setup
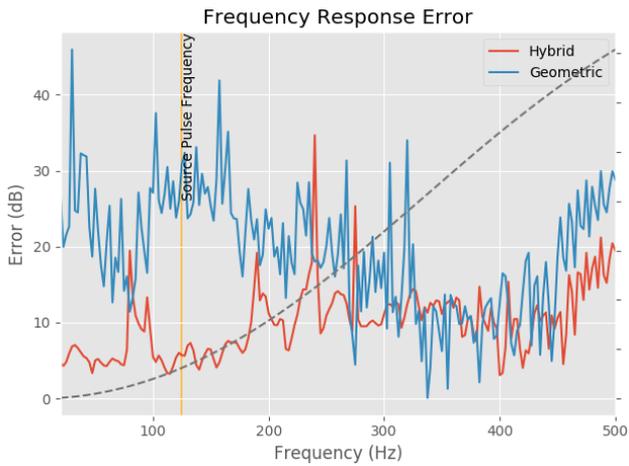
(b) Frequency Response Error

Figure 15: Error comparison with a high resolution numerical simulation on the office scene. (a) shows the experimental setup, with the green region denoting the direct sound outside of which the geometric approach cannot accurately compute. In (b) we compared an impulse response generated by a Gaussian derivative source with a primary frequency of 125 Hz. The yellow line indicates the source frequency while the dotted gray line indicates the transition between the numerical and geometric components of the hybrid approach. The critical frequency of this filter is at 500Hz. In the frequency band centered around the source frequency (the 88Hz to 177Hz band), we noted an average 6 dB error for the hybrid approach, which, while noticeable, is small. On the other hand the geometric approach yielded an average 27 dB error, which is significant.
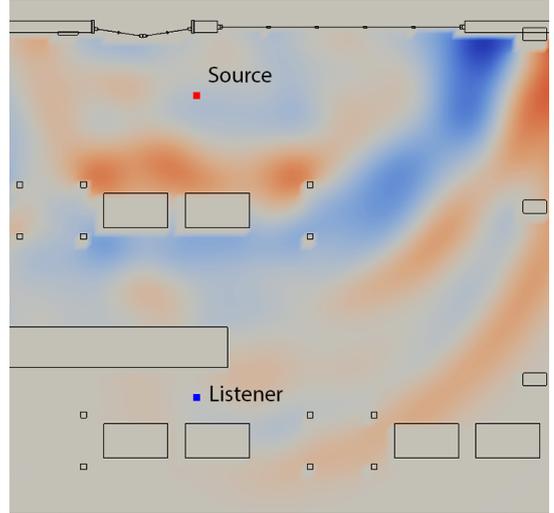
(a) Experimental Setup
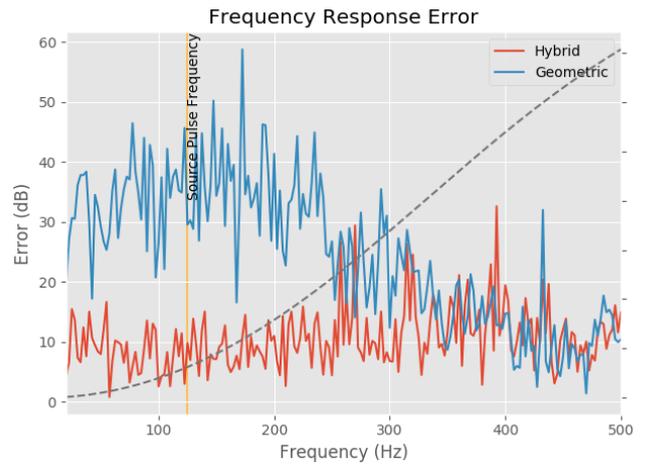
(b) Frequency Response Error

Figure 16: Error comparison with a high resolution numerical simulation on the warehouse scene. (a) shows the experimental setup. In (b) we compared an impulse response generated by a Gaussian derivative source with a primary frequency of 125 Hz. The yellow line indicates the source frequency while the dotted gray line indicates the transition between the numerical and geometric components of the hybrid approach. The critical frequency of this filter is at 500Hz. In the frequency band centered around the source frequency (the 88Hz to 177Hz band), we noted an average 8 dB error for the hybrid approach, which, while noticeable, is small. On the other hand the geometric approach yielded an average 36 dB error, which is significant.

We highlight the performance of our optimization technique on a variety of CAD models and we have evaluated on complex models of office and warehouses. Additionally, our approach is general for use by designers and engineers since it only requires a CAD mesh definition as input.

Our approach has some limitations. We assume that acoustic material characteristics of the environment are available and fixed. The accuracy of the propagation results is a function of these material characteristics. We also assume that the optimization function and the solver can compute the most optimal noise based on equation 6 or acoustic characteristics of the environment. Additionally, our wave-based technique assumes a heterogeneous environment and thus cannot propagate transmission effects. We would like to explore a hybrid scheme that can utilize these effects for noise propagation. In the future, we would like to extend to other types of noise models. We only utilized A-weighting curves. However other measures such as time-weighted average metrics are often used for safety and health purposes. Additionally, some metrics are based on noise rating curves; these test whether or not a frequency spectrum is fully above or below a specific noise curve rating. We would like to evaluate on other complex scenarios, including industrial models corresponding to the factories with heavy machinery. We would also like to explore alternatives to just source placement and also optimize the geometry or obstacle placement within a scene. Finally, we would like to extend to outdoor noise control applications, where it is important to know the occluder or wall locations (e.g. along a highway).

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

[2] Mathias Basner, Wolfgang Babisch, Adrian Davis, Mark Brink, Charlotte Clark, Sabine Janssen, and Stephen Stansfeld. Auditory and non-auditory effects of noise on health. *The Lancet*, 383(9925):1325–1332, 2014.

[3] Alban Bassuet, Dave Rife, and Luca Dellatorre. Computational and optimization design in geometric acoustics. *Building Acoustics*, 21(1):75–86, 2014.

[4] Elliott H Berger. *The noise manual*. Aiha, 2003.

[5] Stefan Bilbao. Modeling of complex geometries and boundary conditions in finite difference/finite volume time domain room acoustics simulation. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(7):1524–1533, 2013.

[6] Berglund Birgitta, Lindvall Thomas, and HS Dietrich. Guidelines for community noise. *World Health Organization, Geneva*, page 21, 1999.

[7] Voichita Bucur. *Urban forest acoustics*. Springer Science & Business Media, 2007.

[8] Jeng-Tzong Chen, Ying-Te Lee, and Yi-Jhou Lin. Analysis of mutiple-shepers radiation and scattering problems by using a null-field integral equation approach. *Applied Acoustics*, 71(8):690–700, 2010.

[9] Robert D Ciskowski and Carlos Alberto Brebbia. *Boundary element methods in acoustics*. Springer, 1991.

[10] Peter D'Antonio and Trevor J Cox. Room optimizer: A computer program to optimize the placement of listener, loudspeakers, acoustical surface treatment and room dimensions in critical listening rooms. In *Audio Engineering Society Convention 103*. Audio Engineering Society, 1997.

[11] Eduard Deines. *Acoustic simulation and visualization algorithms*. PhD thesis, Techn. Univ., Fachbereich Informatik, 2008.

[12] Maria B Dühring, Jakob S Jensen, and Ole Sigmund. Acoustic design by topology optimization. *Journal of sound and vibration*, 317(3):557–575, 2008.

[13] Jean-Jacques Embrechts, Dominique Archambeau, and Guy-Bart Stan. Determination of the scattering coefficient of random rough diffusing surfaces for room acoustics applications. *Acta Acustica united with Acustica*, 87(4):482–494, 2001.

[14] Thomas Funkhouser, Ingrid Carlbom, Gary Elko, Gopal Pingali, Mohan Sondhi, and Jim West. A beam tracing approach to acoustic modeling for interactive virtual environments. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 21–32. ACM, 1998.

[15] S Hampel, S Langer, and AP Cisilino. Coupling boundary elements to a raytracing procedure. *International journal for numerical methods in engineering*, 73(3):427–445, 2008.

[16] Maarten Hornikx, Roger Waxler, and Jens Forssén. The extended fourier pseudospectral time-domain method for atmospheric sound propagation. *The Journal of the Acoustical Society of America*, 128(4):1632–1646, 2010.

[17] IEC. Electroacoustics - sound level meters. IEC 61672-1, International Electrotechnical Commission, Geneva, Switzerland, 2013.

[18] Ahmad Jamrah, Abbas Al-Omari, and Reem Sharabi. Evaluation of traffic noise pollution in amman, jordan. *Environmental Monitoring and Assessment*, 120(1):499–525, 2006.

[19] Jian Kang. *Urban sound environment*. CRC Press, 2006.

[20] Jukka Keränen, Erkko Airo, Pekka Olkinuora, and Valtteri Hongisto. Validity of ray-tracing method for the application of noise control in workplaces. *Acta Acustica united with Acustica*, 89(5):863–874, 2003.

[21] Hanieh Khalilian, Ivan V Bajić, and Rodney G Vaughan. Joint optimization of loudspeaker placement and radiation patterns for sound field reproduction. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 519–523. IEEE, 2015.

[22] Caitlin R Kight and John P Swaddle. How and why environmental noise impacts animals: an integrative, mechanistic review. *Ecology letters*, 14(10):1052–1061, 2011.

[23] Siegfried H Linkwitz. Active crossover networks for noncoincident drivers. *Journal of the Audio Engineering Society*, 24(1):2–8, 1976.

[24] Tapio Lokki, Alex Southern, Samuel Siltanen, and Lauri Savioja. Studies of epidaurus with a hybrid room acoustics modelling method. *Acoustics of Ancient Theaters Patras, Greece*, 2011.

[25] Ravish Mehra, Nikunj Raghuvanshi, Lauri Savioja, Ming C Lin, and Dinesh Manocha. An efficient gpu-based time domain solver for the acoustic wave equation. *Applied Acoustics*, 73(2):83–94, 2012.

[26] Michael Monks, Byong Mok Oh, and Julie Dorsey. Audioptimization: Goal-based acoustic design. *Computer Graphics and Applications, IEEE*, 20(3):76–90, 2000.

[27] A Montazeri, J Poshtan, and MH Kahaei. Optimal placement of loudspeakers and microphones in an enclosure using genetic algorithm. In *Control Applications, 2003. CCA 2003. Proceedings of 2003 IEEE Conference on*, volume 1, pages 135–139. IEEE, 2003.

[28] Nicolas Morales and Dinesh Manocha. Efficient wave-based acoustic material design optimization. *Computer-Aided Design*, 78:83–92, 2016.

[29] Nicolas Morales, Ravish Mehra, and Dinesh Manocha. A parallel time-domain wave simulator based on rectangular decomposition for distributed memory architectures. *Applied Acoustics*, 97:104–114, 2015.

[30] AM Ondet and JL Barbry. Modeling of sound propagation in fitted workshops using ray tracing. *The Journal of the Acoustical Society of America*, 85(2):787–796, 1989.

[31] OSHA. Occupational noise exposure. OSHA 1910.95, Occupational Safety and Health Administration, 2008.

[32] A Piccolo, D Plutino, and G Cannistraro. Evaluation and analysis of the environmental noise of messina, italy. *Applied Acous-*

tics, 66(4):447–465, 2005.

[33] Nikunj Raghuvanshi, Rahul Narain, and Ming C Lin. Efficient and accurate sound propagation using adaptive rectangular decomposition. *Visualization and Computer Graphics, IEEE Transactions on*, 15(5):789–801, 2009.

[34] Philip W Robinson, Samuel Siltanen, Tapio Lokki, and Lauri Savioja. Concert hall geometry optimization with parametric modeling tools and wave-based acoustic simulations. *Building Acoustics*, 21(1):55–64, 2014.

[35] Atul Rungta, Sarah Rust, Nicolas Morales, Roberta Klatzky, Ming Lin, and Dinesh Manocha. Psychoacoustic characterization of propagation effects in virtual environments. *ACM Transactions on Applied Perception (TAP)*, 13(4):21, 2016.

[36] Shinichi Sakamoto, Ayumi Ushiyama, and Hiroshi Nagatomo. Numerical analysis of sound propagation in rooms using the finite difference time domain method. *The Journal of the Acoustical Society of America*, 120(5):3008–3008, 2006.

[37] Kai Saksela, Jonathan Botts, and Lauri Savioja. Optimization of absorption placement using geometrical acoustic models and least squares. *The Journal of the Acoustical Society of America*, 137(4):EL274–EL280, 2015.

[38] Tetsuya Sakuma, Shinichi Sakamoto, and Toru Otsuru. *Computational simulation in architectural and environmental acoustics*. Springer, 2014.

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

[40] Carl Schissler and Dinesh Manocha. Interactive sound propagation and rendering for large multi-source scenes. *ACM Transactions on Graphics (TOG)*, 36(1):2, 2016.

[41] Carl Schissler, Ravish Mehra, and Dinesh Manocha. High-order diffraction and diffuse reflections for interactive sound propagation in large environments. In *Proc. of ACM SIGGRAPH*, volume 33, pages 1–12, 2014.

[42] Martín E Sequeira and Víctor H Cortínez. Optimal acoustic design of multi-source industrial buildings by means of a simplified acoustic diffusion model. *Applied Acoustics*, 103:71–81, 2016.

[43] Alexander Southern, Samuel Siltanen, and Lauri Savioja. Spatial room impulse responses with a hybrid modeling method. In *Audio Engineering Society Convention 130*. Audio Engineering Society, 2011.

[44] Micah Taylor, Anish Chandak, Qi Mo, Christian Lauterbach, Carl Schissler, and Dinesh Manocha. Guided multiview ray tracing for fast auralization. *IEEE Transactions on Visualization and Computer Graphics*, 18:1797–1810, 2012.

[45] Lonny L Thompson. A review of finite-element methods for time-harmonic acoustics. *The Journal of the Acoustical Society of America*, 119(3):1315–1330, 2006.

[46] Nicolas Tsingos, Carsten Dachsbacher, Sylvain Lefebvre, and Matteo Dellepiane. Instant sound scattering. In *Proceedings of the 18th Eurographics conference on Rendering Techniques*, pages 111–120. Eurographics Association, 2007.

[47] Nicolas Tsingos, Thomas Funkhouser, Addy Ngan, and Ingrid Carlbom. Modeling acoustics in virtual environments using the uniform theory of diffraction. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 545–552. ACM, 2001.

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

[49] Ying Wang, Safieddin Safavi-Naeini, and Sujeet K Chaudhuri. A hybrid technique based on combining ray tracing and fdtd methods for site-specific modeling of indoor radio wave propagation. *IEEE Transactions on antennas and propagation*, 48(5):743–754, 2000.

[50] Mianqiang Xue, Yichen Yang, Jujun Ruan, and Zhenming Xu. Assessment of noise and heavy metals (cr, cu, cd, pb) in the ambience of the production line for recycling waste printed circuit boards. *Environmental science & technology*, 46(1):494–499,

2011.

[51] Hengchin Yeh, Ravish Mehra, Zhimin Ren, Lakulish Antani, Dinesh Manocha, and Ming Lin. Wave-ray coupling for interactive sound propagation in large complex scenes. *ACM Transactions on Graphics (TOG)*, 32(6):165, 2013.

[52] Paulo Henrique Trombetta Zannin, Fabiano Belisário Diniz, and Wiliam Alves Barbosa. Environmental noise pollution in the city of curitiba, brazil. *Applied Acoustics*, 63(4):351–358, 2002.