Multi-Resolution Sound Rendering

Michael Wand* Wolfgang Straßer*

WSI/GRIS, University of Tübingen

Introduction

We present a new approximation technique based on hierarchical stochastic sampling that allows sound rendering in highly complex acoustic environments, i.e. scenes containing a large number of sound sources. We apply the algorithm to render scenes with many individual sources (such as a football stadium) and to auralize simulations of global sound propagation.

A lot of work has been done in the area of sound rendering: The main areas of research are simulation of global sound propagation, the synthesis of sound sources themselves and auditory display, i.e. techniques for a faithful sound reproduction through speaker systems. A detailed discussion of the state of the art can be found in survey articles such as [Funkhouser et al. 2002]. However, up to now, only little attention has been paid to scenes containing many sound sources, such as a football stadium or a machine hall containing many thousands of individual sound emitters. A recent algorithm by Tsingos et al. [Tsingos et al. 2003] allows an approximate rendering of several hundred sound sources using dynamic clustering. However, the approach is not yet efficient enough for handling complex scenes such as a football stadium with ten thousand football fans.

We propose a stochastic sampling approach for an approximate rendering of scenes containing very large amounts of sound sources: A sampling algorithm selects a set of representative sound sources according to an importance sampling strategy. Sound sources with larger volume are sampled with higher probability. The algorithm uses a combination of a precomputed spatial hierarchy and a dynamic sampling algorithm, allowing output-sensitive rendering times, i.e. running time independent of the input complexity. Using a prototype implementation of the algorithm, we show that renderings of approximations of acceptable quality are possible in real-time.

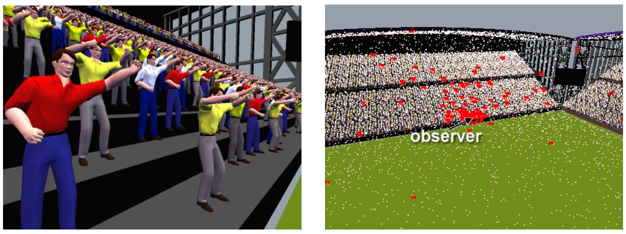
The Sampling Algorithm

Our rendering system consists of two parts: A sampling thread that selects representative sound sources, and a mixing thread that blends together lists of sound sources obtained from the sampler. The sampling algorithm consists again of two parts: a dynamic sampling algorithm and a hierarchical algorithm that delivers candidate sets for dynamic sampling.

Dynamic Sampling: The dynamic sampling algorithm examines a list of sound sources and computes the volume of each sound source. The volume is a product of (1) the original volume of sound source itself, (2) the attenuation due to the distance to the observer, and (3) the directional characteristic of the emitter and the receiver. The algorithm chooses a set of k representative sound sources for a user defined constant k. Each sound source is included in the sample set with probability proportional to its volume. If the position of the observer changes, the sample set is updated dynamically: If the volume of a sound source changes by more than a constant factor $1 \pm \varepsilon$, it is replaced by a new random sample according to the current importance distribution. The sample sets are handed over to the mixer thread at an update rate of typically 20Hz. Sources that are replaced are faded in and out smoothly in order to avoid discontinuity artefacts.

Candidate Sets: The dynamic sampling algorithm has a linear time complexity, precluding interactive rendering of complex sound environments. Thus, we add a step prior to dynamic sampling that determines a candidate set of reduced complexity. This algorithm uses an octree that has been build during preprocessing. The octree stores the original sound sources in its leaf nodes. Each inner node contains one representative sound source that has been selected from the sound sources stored in its direct children with a probability proportional to their volume. The volume of the representative source is set to the sum of the child volumes. Using this hierarchy, we can perform importance sampling according to the volume (1) and distance (2) of the sound sources in fully output-sensitive time, independent of the number of sound sources in the scene. To account for (at least semi-diffuse) directional emission effects (3), we use a candidate set that is larger (say 10×) than the desired sample size k and apply the dynamic selection algorithm to which prefers sound sources that emit in the direction of the observer.

{wand, strasser}@gris.uni-tuebingen.de



Example: *Football Stadium. Right: Sample sources distribution.*

Results

We have employed the sound rendering algorithm for two different application scenes: The first is a football stadium containing 16,000 football fans, singing and shouting. We have rendered an approximation using 2,000 sound sources that provided an acceptable rendering quality, as confirmed by a preliminary user study. This number of sound sources is also the theoretical limit for real-time mixing using DirectX software sound buffers (the accompanying video was recorded using simple unoptimized C++ mixing, which is slower). Fewer sound sources also already yield believable results but they can still be easily distinguished from the original sound (i.e. rendering by mixing all sources).

The second example is a hall with a violin player. A ray-tracing based acoustics simulation was used to compute 50,000 phantom sound sources that approximate global reverberations. Frequency dependent directional reflection was modeled by splitting the original samples into frequency bands and specifying varying directional emission characteristics to the phantom sources. Our sampling algorithm is then applied to reduce the complexity and render interactive walkthroughs. Similar parameters as in the first example yielded plausible renderings of the simulation.

References

[Funkhouser et al. 2002] FUNKHOUSER, T., JOT, J., and TSINGOS, N.: "Sounds Good to Me!", Computational Sound for Graphics, Virtual Reality, and Interactive Systems. In: *SIGGRAPH 2002 Course Notes*.

[Tsingos et al. 2003] TSINGOS, N., GALLO, E., and DRETTAKIS, G.: Breaking the 64 spatialized sources barrier. In: *Gamasutra Audio Resource Guide* (`www.gamasutra.com`), May 2003.