MUSIC 159: Mid-term Project Proposal

Hwi Joo (Peter) Park

November 12, 2024

Abstract

In this following paper, I will explore various algorithms (Granular Synthesis and Convolution Cross-synthesis) learned in this class that will be used for my final project. After describing them, I will suggest a novel approach to sound design by combining elements of each algorithm to create a unified algorithm (which I call GCS). I will discuss why this new approach is more versatile than the current algorithms. Finally, I will discuss the implications and examples of how the new algorithm can be used in different musical settings.

1 Basic Algorithms for Digital Sound Design

1.1 Granular Synthesis

1.1.1 Introduction to Granular Synthesis

Granular Synthesis (GS) is a sound design technique that uses grains (usually 10∼100ms samples) to generate sounds. It is popular because compared to other modulation techniques, like Additive and Subtractive synthesis, it is able to generate complex and diverse sounds relatively conveniently by tweaking the various parameter values. Furthermore, it is able to take arbitrary waveforms, such as pre-recorded audio samples.

GS is used in even the most basic functionalities of DAWs. Time-stretching–increasing the length of an audio sample without modifying pitch–is made possible by GS. The analogue, Pitch-shifting–altering pitch without changing the speed of the audio sample–is also a prime example. The following subsections will deal with the construction and organization of grains that make GS the way it is.

1.1.2 Grain Parameters

Granular Synthesis begins with the generation of the grain, which serves as the basic waveform. The starting parameters are retrieval_point (starting point), grain_duration, playback_speed. Note that since end_point = retrieval_point + grain_duration, we can obtain a grain from an audio sample by simply cropping it around the start and end point.

1.1.3 Grain Envelopes

In order to layer the grains over one another, we must apply amplitude envelopes or windows on the grain we've constructed. Without such amplitude modifications, artifacts or unnatural amplitude shifts caused by abrupt transitions may occur. The default envelope used are Gaussians, but complex shapes can give us more control over the overall shape of the grain.

1.1.4 Grain Organization

Grains can be ordered in multiple ways. Grain density–defined by the number of grains in one second–serves as a numeric metric to describe density/sparsity of sound. Depending on the grain density, the overlapped grains may sound rhythmic or continuous.

Synchronous grain synthesis involves using a constant grain delay, which means every grain is spaced equally apart. When the density is low, we can expect a rhythmic effect. However, as periodicity increases, a inversely correlated frequency is formed.

Asynchronous grain synthesis is similar, but instead of using a constant grain delay, we randomize it. Most parameters we have previously mentioned can be randomized (ie. retrieval_point, grain_duration, playback_speed) by making additional random parameters that are multiplied by a noise factor. This leads to a noisier and more dynamic sounds.

1.2 Convolution-based Cross-synthesis

1.2.1 Introduction to Cross-synthesis and Convolution

Cross-synthesis is the process of transferring some characteristics of one audio sample onto another. Cross-synthesis can be implemented in many ways, such as by convolution or by matching pursuit algorithms–both covered in lecture. *Convolution* is an operation that is defined as the integral of products of two functions, after one is reflected across the y-axis and shifted.

In Audio DSP, convolution is often applied to make *convolution reverbs*. By convolving a sample with an impulse response file, we bring that sample into a new environment, which creates the reverb provided by the impulse. For example, a gun-fire sample convolved with a clap in a concert hall (impulse response file) would emulate a the sound of a gun-fire in a concert hall.

1.2.2 The Mathematical Formulation of Convolution

In image processing, the convolution operation involves a two-dimensional filter that slides over the input image. This process allows the extraction of similarity and patterns between the input image and the filter (ie. a corner edge filter will have high similarity with corners, leading to high intensities in those areas for the resulting image).

The same can be said of audio. Note that since convolution is commutative, we can think of it as either the impulse sliding over at each step and multiplying the input signal, or the other way around. Either way, the results are equivalent. Since digital audio is discrete, we use a summation instead of integrals.

First, we observe that convolution of an input signal with the Kronecker-delta function results it the signal itself:

$$x(nT) = \sum_{k=-\infty}^{\infty} x(kT)\delta(nT - kT) \quad \text{where} \quad \delta(nK - nT) = \begin{cases} 1 \text{ if } n = k, \\ 0 \text{ if } otherwise. \end{cases}$$

In the above function, we note that $x(nT)$ is the input signal. This function outputs the amplitude at $time = nT$. $T = \frac{1}{sampling\_rate}$ and n is the n^{th} sample of the input signal, which is also, in array notation, $x[n]$.

Now, substituting the delta function with the impulse response (notated as h):

$$y(nT) = \sum_{k=-\infty}^{\infty} x(kT)h(nT - kT)$$

In conclusion, this equation can be thought of as sliding a flipped version of one audio signal across another, multiplying overlapping values at each shift, and summing the results to produce each output value.

2 Granular Convolution Synthesis (GCS)

2.1 The Intuition

While granular synthesis and cross-synthesis convolution are fundamental sound design techniques, they have limitations. GS, while able to generate diverse textures and ambient sounds, often times retains the original signature of the "grain" itself. In other words, it has limited spectral complexity. While we used windows to reduce artifacts, a bad set of parameters can still generate sounds with much artifacts.

The good news is that GCS may alleviate many of the limitations. Convolution combines two sounds together, leading to increase spectral complexity. Convolution is also smoothing operator, which means the issue with artifacts may be resolved.

Convolution is also not without limitations. First of all, the effects of convolution is only as good as its impulse response. Convolution also doesn't change the temporal complexity, nor the dynamics of pitch. Granular synthesis is ideal for time-stretching and pitch-shifting, with further temporal control based on how the grains are chosen to be organized. This makes the two algorithms combined a convincing and powerful sound design tool.

2.2 The Algorithm

Algorithm 1 Granular Convolution Synthesis (GCS)

1: **Input:** Audio Clip $\mathbf{A}$, Impulse Response $\mathbf{h}$, *parameters*: retrieval_point, grain_duration, playback_speed, pitch_shift, window_type
2: **Output:** Resulting Audio $\mathbf{y}$
3: **Generate Sample:** Let grain, $\mathbf{x} = \mathbf{A}$[retrieval_point:retrieval_point + grain_duration]
4: **Convolution:** Convolve the grain $\mathbf{x}$ with impulse $\mathbf{h}$, let the result be $\mathbf{y}$
5: **Windowing:** Apply window_type (ex. Hanning, Gaussian) window on $\mathbf{y}$
6: **Organize Grains:** Apply add-overlap to generate grain organization

3 Practical uses of GCS in Musical Settings

3.1 Environmental Sound Modeling

The GCS, with the granular and convolution component will be ideal for simulating sounds in movies or games. Instead of using the same sound effects for all locations in the game, video game sound designers can choose grains (sound effects) convolved with location-specific impulses to generate location-particular sound effects. If the default sound effect is already ready, and we know there are various locations in a game, then by passing through GCS, we can simulate realistic sound effects based on the user's location in the game/movie-world.

This would make producing complex sounds with diverse ambiances and control over temporal components easier–simply by turning knobs and providing a starting audio sample. It can also make the movie/game more dynamic by generating ambiances corresponding to each location the player might be in, providing a responsive experience.

3.2 Live Performance Sound Design Tool

Like in the previous example, if the few parameters of GCS can be mapped to a physical device, it would be very easy to physically turn the knobs and generate new audio on the spot. This can be ideal for generating experimental and new sounds. Artists can feed live input (ie. voice) to apply various reverbs and grain positioning, pitch-shift and time-stretches as necessary. Instead of having to move back and forth between two separate algorithms, an artist can use one device to save templates and presets to run GCS instantly on stage.