Perception and Multimedia Computing

Additive synthesis, reverb, and aliasing

Friday 15th Dec 2017

1. Additive synthesis

A square wave has the following shape:

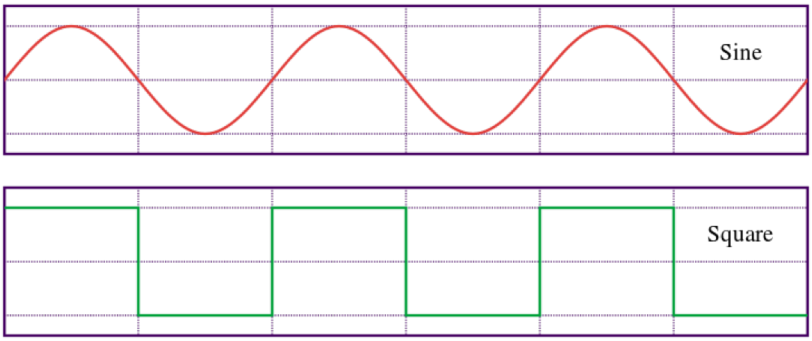

Figure 1: Source: https://en.wikipedia.org/wiki/Square_wave#/media/File:Waveforms.svg

The Fourier Theorem tells us that we can perfectly re-create a square wave using a sum of sine waves (if we carefully choose their frequencies, amplitudes, and phases).

Specifically, we can create a square waves using a sum of sines, using the following equation

$$square(t) = \frac{4}{\pi}\sum_{k=1}^{\infty}\frac{\sin(2\pi(2k-1)ft)}{2k-1}$$

where *square(t)* is the value of the wave at time *t*.

(If it's been a while since you used a summation sign, you might refresh yourself here: http://www.columbia.edu/itc/sipa/math/summation.html)

a. Recall that the generic equation for a sine wave at time *t* is:

$$f(t) = A\ \sin(2\pi ft + \varphi)$$

i. What is the frequency of the first sine wave in the square wave summation (i.e., the sine wave that corresponds to *k*=1 in the top equation)?
ii. What is the amplitude of the first sine wave?
iii. What is the phase of the first sine wave?
iv. What is the frequency of the second sine wave (i.e., the sine wave that corresponds to *k*=2)?
v. What is the amplitude of the second sine wave?
vi. What is the phase of the second sine wave?
vii. What is the frequency of the *k*-th sine wave? (i.e., express the frequency as a function of *k*)
viii. What is the amplitude of the *k*-th sine wave? (i.e., express the amplitude as a function of *k*)
ix. What is the phase of the *k*-th sine wave?

b. **Now edit the additive.js sketch (download it from learn.gold)** to implement the creation of a square wave using a finite number of sine waves. Specifically, fill in the for-loop in the `setup()` function so that the first `numHarmonics` sines are generated and played. (Note that playing them simultaneously is the same thing as adding them together, in terms of the waveform output by your computer.) Then, answer the following questions:

   i. What does the waveform look like when `numHarmonics` = 1?
   ii. How does the waveform shape change as the number of harmonics increases?
   iii. How does the sound change as the number of harmonics increases?
   iv. How does the sound change as the fundamental frequency increases?
   v. What happens to the sound when you use a fundamental frequency of around 1000Hz and a very large number of harmonics (e.g., 500)?

2. Intro to reverb using convolution:

In lecture, we saw how you can apply reverb to a sound using the following steps:

- Step1: Record an *impulse response* in a space (i.e., record an "impulse" sound like a clap or air gun, using a microphone placed in some position in the space).

- Step2: *Convolve* that impulse response recording with a new sound, to make it sound like the sound took place in the same acoustic environment where the impulse response was recorded.

In this part of the lab, you will try the p5 convolution function using a pre-recorded impulse response and any other sound of your choice.

- Download a set of free impulse responses from http://www.voxengo.com/impulses/

- Use Audacity to record a short audio file (1-2 seconds, for instance you speaking a few words).

- Download the folder "p5Convolve_skeleton" and load the two sound files, and listen to the result. How does the sound of the impulse response relate to the sound of its reverb applied to a new sound? Try different impulse responses.

3. Aliasing

Use the sineKeyboard and/or sineMouse sketches to explore the following questions.

a. These sketches run with a sampling rate of 44100Hz. What is the Nyquist frequency?
b. What frequency will you hear if you synthesise a sine wave at 10,000Hz? Why?
c. What frequency will you hear if you synthesise a sine wave at 34,100Hz? Why?
d. What frequency will you hear if you synthesise a sine wave at 43,100 Hz? Why?
e. Edit the SineWaveMouse sketch so that the horizontal mouse position is mapped from 0Hz at the left-most edge of the screen to 100,000Hz at the right-most edge of the screen. You'll need to edit the line:

```
freq=map(mouseX, 0, width, 20.0, 20000.0);
```

f. What do you expect the pitch to do as you move from 0Hz to 22,050Hz? Why? Verify that the sketch makes the sound you expect.
g. What do you expect the pitch to do as you move from 22,050Hz to 44,100Hz? Why? Verify that the sketch makes the sound you expect.
h. What do you hear when you move from 44,100Hz to 88,200Hz? How does this compare to what you hear when moving from 0Hz to 44,100Hz?
i. What does this suggest about aliasing when your frequency is above the sample rate?