Web Audio Codelab #4

Sample, Filter, and Delay

Music 220A Summer 2021
CORS: Cross-Origin Resource Sharing

ABC.COM

index.html
image.jpg
sound.wav

XYZ.COM

bam.wav
AudioBuffer

ABC.COM

sound.wav

fetching

ArrayBuffer

(decoded byte array)

AudioBuffer

(Float32Array)
const response = await fetch(url);
const arrayBuffer = await response.arrayBuffer();
const audioBuffer = await audioContext.decodeAudioData(arrayBuffer);
AudioBufferSourceNode

AudioBuffer

(Float32Array)
const bufferSource =
    new AudioBufferSourceNode(audioContext, {buffer: audioBuffer});
const amp = new GainNode(audioContext);
bufferSource.connect(amp).connect(audioContext.destination);
bufferSource.start();
MediaStream

ABC.COM

sound.wav

AudioElement

(streaming)

(MediaStream)

(Float32Array)
```javascript
const audioElement = new Audio('./media/loop-1.wav');
audioElement.crossOrigin = 'anonymous';
audioElement.addEventListener('canplaythrough', () => {
  const mediaStream = audioElement.captureStream();
  const streamSource =
    new MediaStreamAudioSourceNode(context, {mediaStream: mediaStream});
  streamSource.connect(context.destination);
  audioElement.play();
});
```
AudioBufferSourceNode

- (AudioParam) playbackRate
- (AudioParam) detune
Granular Synthesis

Time granulation of sampled sounds

process sounds through a logical trash machine...

reorder

merge/reorder

replicate

Copyright Fernando Lopez-Loreano 1997
Granular Synthesis

- synthetic: sum of sines
- sampled: part of a soundfile...

Waveform

Amp Env

Output

Copyright Fernando Lopes-Lezcano 1997

Nando’s slide
Delay

\[ \text{Delay} \quad (t) \]
Delay + Feedback

Delay \((t)\)

Feedback (gain = 1)
Delay + Feedback

Delay $(t)$

Feedback (gain < 1)
Delay + Feedback

![Diagram showing a delay (\(t\)) and feedback (gain > 1)]
Delay + Feedback + Modulation

![Diagram showing delay, feedback, and modulation with labels for LFO, delay time, and feedback.](image-url)
# Delay effect range

Table 7. Approximate effect delay range in milliseconds.

<table>
<thead>
<tr>
<th>Effect</th>
<th>Onset</th>
<th>Nominal</th>
<th>Range End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vibrato$^{83}$</td>
<td>0</td>
<td>Minimal</td>
<td>5</td>
</tr>
<tr>
<td>Flange</td>
<td>0</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>Chorus</td>
<td>1</td>
<td>5</td>
<td>30</td>
</tr>
<tr>
<td>Doubling</td>
<td>10</td>
<td>20</td>
<td>100</td>
</tr>
<tr>
<td>Echo</td>
<td>50</td>
<td>80</td>
<td>$\infty$</td>
</tr>
</tbody>
</table>

From: https://ccrma.stanford.edu/~dattorro/EffectDesignPart2.pdf
Examples:

Sample, Filter, and Delay

- Looping fun
- Simple drum sampler
- Roland TR808 on Web
- Filters in the real world
- Filter sweeping
- Delay-based Effects (source)