INTRODUCTION TO COMPUTER MUSIC
SPECTRAL INTERPOLATION SYNTHESIS

Roger B. Dannenberg
Professor of Computer Science, Art, and Music

SPECTRAL INTERPOLATION SYNTHESIS

Roger Dannenberg
Marie-Helen Serra
Dean Rubine
Istvan Derenyi
Hank Pelerin
Ning Hu
Additive Synthesis and Table Lookup Synthesis

• Additive Synthesis:
  • Every partial has independent frequency and amplitude
  • Analysis/synthesis possible, but no simple parametric control

• Table-Lookup Synthesis
  • Relative amplitudes of all partials are locked in
  • Frequencies are all harmonic

Table Lookup Oscillator Review

```c
float table[513] = { ... some waveform ... };  
double phase = 0.0;  
void osc(double hz, float table[], float out[]) {
    double incr = hz * 512 / sample_rate;
    for (int i = 0; i < block_size; i++) {
        int iphase = floor(phase);
        double x1 = table[iphase];
        out[i] = x1 + (phase – iphase) * (table[iphase+1] – x1);
        phase += incr;
        if (phase > 512) phase = phase – 512;
    }
} 
```
Spectral Interpolation

- Interpolate between tables
- Keep phases coherent so that interpolation is truly interpolation of spectra
- Restricted to harmonic spectra

Spectral Variation by Interpolation

- Reload tables with new spectra
- Relatively slow update (~20 tables/second)
- Where do tables come from?
Use Pitch and Amplitude to Compute Spectra

Measuring Spectra

- We use phase-vocoder style analysis on
- Crescendo at different pitches
- Yields 2-D (pitch, amplitude) tables of spectra
Attacks Are Too Rapid And Inharmonic

- Solution: use sampled attacks (~30ms)
- New Problem: How do you join attack to synthesized sound?
  - Cross-fade does not work – too many phase problems
  - Make attack long enough to settle to harmonics
  - Analyze phase of every partial at end of attack
  - Synthesize tables with matching phase
  - Splice with very short cross-fade (2ms) or none

WHERE DOES CONTROL INFORMATION COME FROM?
SIS Research Approach

Score

Control

Synthesis Algorithm

Sound

SIS Research Approach

Score

Control

Synthesis Algorithm

Sound
Divide-and-Conquer

1. Score
   - Apply Performance Model
2. Control Signals
   - Apply Synthesis Model
3. Audio

Research Model: Synthesis Refinement

1. Score
   - Human Performance
2. Audio
   - Parameter Extraction
3. Control Signals
   - Apply Instrument Model
4. Audio
   - Human Comparison, Refine Instrument Model
Introduction

• Envelopes:
  • are crucial to sound synthesis;
  • depend upon context.
• We claim:
  • envelopes largely determined by context;
  • envelope generation techniques can improve synthesis.

The Big Picture
Related Work

- Moorer, Grey, and Strawn ‘78
- Clynes ‘85
- Chafe ‘89
- Canazza, De Poli, Roda’, and Vidolin ‘97
- Sundberg, Askenfelt, and Fryden ‘83
- Key: consider context, study envelopes in musical phrases

An Experiment

Question: How do factors affect the center of mass* (first moment)?

*Why Center of Mass?: simplest measure of shape.
What Did We Measure?

- Notes and envelopes defined by inter-onset times.
  - because end-of-note not well defined
  - “micro pauses” incorporated into data

Some Results

- Up-up phrases showed later center of mass than other combinations
- Large intervals had earlier center of mass than small intervals
- Legato articulation gave later center of mass than others (this should be obvious).
BACK TO SPECTRAL INTERPOLATION SYNTHESIS

Looking at “Real” Envelopes

Normalized RMS Amplitude vs. Normalized Time
Tongue and Breath

Envelope of a Slurred Note
Envelope of a Slurred Note

Normalized RMS Amplitude vs. Normalized Time

Toward an Envelope Model

- Breath envelope gives overall shape
- Tongue articulates beginning and ending
Amplitude Envelope Specification

Many shapes approximated by a "general" envelope function:

Computing Parameters

• Previous slide shows 9 detailed parameters
• Compute them from:
  • pitch (in semitones, according to score)
  • dur (in seconds, according to score)
  • begin-phrase (is this the first note in a phrase?)
  • end-phrase (is this the last note in a phrase?)
  • from-slur (is there a slur from preceding note?)
  • to-slur (is there a slur to the next note?)
  • direction-up (is this note higher than preceding note?)
Method

• Score provides “actual” parameters
• Study real performances
• Performance provides envelopes
• Find detailed parameters manually
• Generalize from observed trends

Example: $tf$

\[
\begin{align*}
\text{cond} & \text{ (from-slur} \\
& (\text{setf takefrom (if direction-up 0.1 0.04))} \\
& (t \\
& (\text{setf takefrom (+ 0.03 (* 0.01 (- log-dur)))))
\end{align*}
\]

<table>
<thead>
<tr>
<th>$tt$</th>
<th>$tf$</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>$t$</th>
<th>0.1</th>
<th>0.04</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f$</td>
<td>0.03 - 0.01 × log2(dur)</td>
<td></td>
</tr>
</tbody>
</table>
Example: \((st, pr)\)

\[
\text{end-phrase} \rightarrow (0.02, 0.02)
\]
\[
to\text{-slur} \& \text{direction-up} \rightarrow (0.03, \text{dur} \times 0.2)
\]
\[
to\text{-slur} \rightarrow (0.02, 0.001)
\]
\[
o.w. \rightarrow (0.06 + 0.005 \times \log_2(\text{dur}),
0.03 - 0.01 \times \log_2(\text{dur}))
\]

Envelopes from Score

- We use a 9-parameter envelope model
- One “reference” breath envelope is cropped and stretched (2 parameters + duration)
- Breath envelope is multiplied by attack and decay envelopes to derive final envelope
- Envelope depends on: pitch, pitch context, articulation, next note articulation, duration.
- Score-to-envelope mapping is hand-crafted.
Frequency Envelope

- Use analyzed frequency envelopes
- Capture natural frequency variation
- Pitch dependent
- No vibrato (yet)

Sound Examples

- CSIS Synthesis Example
- Another CSIS Synthesis Example
- CSIS (Without Accompaniment)
Summary

• Modeled from “real” performance
• Phrase-at-a-time
• Melodic direction (lookahead)
• Articulation (lookahead)
• Duration
• Pitch
• Currently rule-based, ad-hoc

Conclusions

• Envelopes are critical to music synthesis
• Statistically valid relationships between score parameters and envelope shape
• Breath + Tongue Model
• *Study of musical phrases and notes in context is critical to future synthesis research.*
Implementation

- SNDAN (Beauchamp et al.) for spectral, RMS, and frequency analysis
- Nyquist (www.cs.cmu.edu/~rbd)
  - score representation
  - compute envelope parameters
  - cut, splice, shape envelopes
  - amplitude, frequency curves → spectral interpolation synthesis

So What?

- Based on these examples examples (and others), I claim we have an adequate “Synthesis Algorithm.”
- What happens if we return to the “Traditional Synthesis Research” model with the CSIS synthesis algorithm?
Traditional Synthesis Research

Control → Synthesis Algorithm → Sound

Score → Sound

Compare

SIS With “Standard” Envelopes

- Amplitude, Time, Duration are correct
- Envelope shape is “correct”
- Timbre is correct
- Two Envelopes (better slurs)

- SIS model with computed envelopes
Discussion

• Since the only “wrong” ingredient is envelope shape, envelope shape must be critical for synthesis.
• This observation could only be made after we perfected (?) trumpet synthesis.

More Discussion

• It follows from our results that… Previous work on synthesis was wrongly assuming simple envelope templates are sufficient.
• In other words… If you had a good synthesis algorithm, how would you know it?
Still More Discussion

• Equal time for the physical modelers:

My goal is to develop sufficiently rich instruments that musicians will want to learn how to control them.
(paraphrasing Julius Smith)