MARE 502, Advanced Topics in Computer Music

Dr Torsten Anders
Prof E R Miranda
Interdisciplinary Centre for Computer Music Research (ICCMR)
University of Plymouth
http://cmr.soc.plymouth.ac.uk/

22 October 2009

Outline of this Lecture

Recapitulation

Questions

Recapitulation

Questions

- What definition does Csound need to produce sound?

Recapitulation

Questions

- What definition does Csound need to produce sound?
- Which statements does a Csound orchestra header contain (examples)?

Recapitulation

Questions

- What definition does Csound need to produce sound?
- Which statements does a Csound orchestra header contain (examples)?
- Explain roughly what the following code snippet does.

```
a1, a2 loscil 20000, 1, 1, 1
```

Recapitulation

Questions

- What definition does Csound need to produce sound?
- Which statements does a Csound orchestra header contain (examples)?
- Explain roughly what the following code snippet does.
  ```
  a1, a2 loscil 20000, 1, 1, 1
  ```
- Which parameters (p-fields) are required in a Csound note statement?

Score: Carry I

Shortcuts exist to reduce the typing for a score.

- **.** repeat previous value in same column (whitespace does the same)
- **+** add duration to previous start time (only for start times)
- **<** interpolate between values in same column

Score: Carry II

```
i 1   0    .5         100
i .   +
i .
```

is equivalent to

```
i 1   0          .5         100
i 1   .5         .5         100
i 1   1          .5         100
```

Score: Carry III

```
i1   0   1   100
i1   1   1   <
i1   2   1   <
i1   3   1   400
```

is equivalent to

```
i1   0   1   100
i1   1   1   200
i1   2   1   300
i1   3   1   400
```

Score: Tempo Control

Syntax

```
t start1 tempo1 ... startN tempoN
```

Example

tempo.csd

P-Fields

- Arbitrary csound note parameters (p-fields) can be read by an instrument definition

P-field variable syntax in orchestra

pN, where N is the parameter position in the note statement

Example

pfields.csd

Expressions

For convenience, many mathematics operations are supported with expression instead of opcode syntax

Example

```
aout oscil p4, p5*10 - 50, 1
```

Signal Rates

Csound generates signal types at different rates, e.g.,

- **a-rate**: audio at sampling rate
- **k-rate**: control signals at control rate (e.g., envelopes) running these at lower than audio rate is more efficient
- **i-rate**: values set at init time

The first letter of a variable name indicates its signal type.
Examples: `aout`, `kamp`, `iwave`, `p3` (p-fields are i-rate)

Examples

signalRates.csd, signalRates2.csd

Controls in QuteCsound

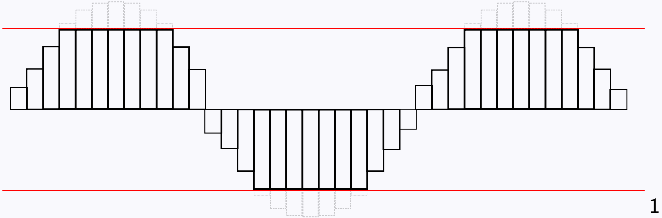
- QuteCsound provides widgets such as sliders, buttons, etc.
- Communication between Csound and these widgets happens via the opcodes `invalue` and `outvalue`

Example

QuteCsoundControls.csd

Clipping I

- Digital audio puts a hard limit on the maximum amplitude possible, which depends on the bit depth.
  - Example: for 16 bit, maximum amplitude is +/- 32767
- Samples are truncated when exceeding the maximum amplitude, resulting in a harsh distortion



Clipping II

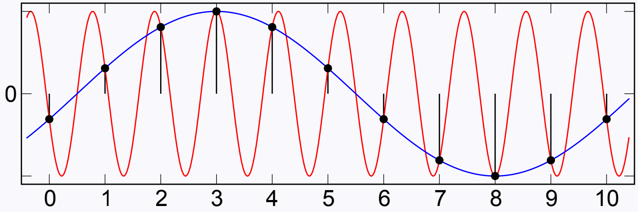
Example

clipping.csd

[1] Source: Wikipedia, Clipping (audio)

Aliasing I

- With digital audio, we can only play frequencies up to half of the sampling rate (Nyquist frequency)
- Higher frequencies result in "under-sampling" – a lower frequency is actually perceived
- Aliasing: frequencies are mirrored at the Nyquist frequency



2

Aliasing II

Examples

aliasing.csd, aliasing2.csd

[2] Source: Wikipedia, Aliasing

Additive Synthesis

- Synthesis technique based on summation of elementary waveforms to create more complex waveforms.
- One of the oldest and most researched techniques

Harmonic spectrum Summation of sine waves whose frequencies are integer multiples of a fundamental.

Inharmonic spectrum Sine wave frequencies are not integer multiples of a fundamental.

Example: individual partial are raised/dampened by sliders

Additive.csd

Example: individual partial are raised/dampened by envelopes

additive1-envs.csd, additive2-envs.csd, additive4-envs.csd

Compositional Application: Stockhauses' Studie II (1954)

- Composition created exclusively from sine waves (in 5-tone equal temperament)
- Score section `http://home.swipnet.se/sonoloco2/Rec/Stockhausen/03.html`

Additive Synthesis Summary

- Advantage: very detailed control over sound – hence enthusiasm for this technique in early days of electronic music
- Disadvantage: much control data to enter
- Addressing disadvantage by retaining advantage: somehow automate control data creation (e.g., by analysis)

Multiple Wavetable Synthesis

- Variant of additive synthesis
- Reducing the total amount of data: using more complex spectra instead of sines
- Individual spectra (static spectrum per octave) won by analysis

References

- Andrew Horner, Lydia Ayers (2000). Continuous-Group Wavetable Synthesis of the French Horn in Csound. In Richard Boulanger (ed.), The Csound book. MIT. (preview available at books.google.com)
- Andrew Horner, Lydia Ayers (2002). Cooking with CSound, Volume 1. A-R Editions. (preview available at books.google.com)

Summary

- Csound basics
    - Carry
    - Tempo
    - Using p-fields
    - Signal rates
    - Controls in QuteCsound
- Digital audio theory
    - Clipping
    - Aliasing
- Additive synthesis
- Multiple wavetable synthesis

Exercises for Next Week

- Read pp. 18-33 of Csound book (Etudes 2-3) available online at `http://www.csounds.com/chapter1/index.html`
- Do your own selection of exercises for etudes 2-3: select tasks that sound interesting for you