Higher Order Integrated Wavetable Synthesis

Andreas Franck
Vesa Välimäki

Fraunhofer Institute for Digital Media Technology IDMT
Ilmenau, Germany

Department of Signal Processing and Acoustics, Aalto University
Espoo, Finland

15th International Conference on Digital Audio Effects DAFx-12
York, September 19, 2012
Outline

Wavetable Synthesis

Higher Order Integrated Wavetable Synthesis

Performance Evaluation

Summary
Wavetable Synthesis

- Widely-used sound synthesis technique
- General idea
  - Store sound in lookup table
  - Single period or longer sounds
  - Phase increments controls pitch
  - Resampling: Arbitrary resampling (ASRC)
- Advantages
  - Efficient synthesis
  - Complex, spectrally rich sounds
Wavetable Synthesis

Decreasing the pitch

- Corresponds to sample rate increase
- Imaging errors
- Fixed-frequency lowpass filter $f_c = f_i / 2$
  - Limits spectral contents
- Multiple wavetables for wide range of pitches

$$|H_c(f)|$$
Wavetable Synthesis
Decreasing the pitch

- Corresponds to sample rate increase
- Imaging errors
- Fixed-frequency lowpass filter \( f_c = f_i / 2 \)
- Limits spectral contents
- Multiple wavetables for wide range of pitches

\[ |H_c(f)| \]
Wavetable Synthesis
Increasing the pitch

- Corresponds to sample rate decrease
- Aliasing errors (in addition to imaging)
- Pitch-dependent lowpass filter $f_c = f_o/2$
  - Rich spectral contents
  - Fewer wavetables necessary
  - More challenging than anti-imaging

\[ |H_c(f)| \]
Wavetable Synthesis
Increasing the pitch

- Corresponds to sample rate decrease
- Aliasing errors (in addition to imaging)
- Pitch-dependent lowpass filter $f_c = f_o/2$
- Rich spectral contents
- Fewer wavetables necessary
- More challenging than anti-imaging

$|H_c(f)|$

$\frac{f_o}{2}$

$\frac{f_i}{2}$

$f$
Higher Order Integrated Wavetable Synthesis
Starting Point: Integrated Wavetables

- Proposed by G. Geiger [1]
- Based on differentiated parabolic wave algorithm (DPW) [2]
- Algorithm
  - Integrate sound before storing in wavetable
  - Differentiate after table lookup

\[ s[n] \xrightarrow{\text{Integrator}} \text{Wavetable} \xrightarrow{H_c(j\Omega)} \text{Resampling Differentiator} \xrightarrow{D(e^{j\omega})} y[m] \]


Higher Order Integrated Wavetable Synthesis

Example: Sawtooth Wave

Ideal bandlimitation
Higher Order Integrated Wavetable Synthesis

Example: Sawtooth Wave

Trivial sampling
Higher Order Integrated Wavetable Synthesis
Example: Sawtooth Wave

Integrated wavetable synthesis [Geiger]
Higher Order Integrated Wavetable Synthesis

- Extension to higher integration/differentiation orders
- Motivated by differentiated parabolic waveform algorithm [1]

\[ D^K(e^{j\omega}) \]

\[ y[m] \]

**Three main components**
- Integrators
- Resampling filter
- \( K \)th order differentiator

Higher Order Integrated Wavetable Synthesis

- Extension to higher integration/differentiation orders
- Motivated by differentiated parabolic waveform algorithm [1]

\[ D^{(K)}(e^{j\omega}) \]

Output

\[ y[m] \]

Three main components

- \( K \) integrators
- Resampling filter
- \( K^{th} \) order differentiator

Higher Order Integrated Wavetable Synthesis

Integrator Design

- Performed at design time (no runtime cost)
- Discrete summation
- Determine constants of integration
  - Make waveform periodic
  - Remove DC before integration
  - See paper
Higher Order Integrated Wavetable Synthesis
Resampling Filter

- Interpolation between wavetable entries
- Provide anti-imaging
- [Geiger]: “Round to nearest” (order $N = 0$)
- Here: Lagrange interpolation
  - Good quality at low frequencies
  - Efficient implementations
  - Orders $N = 1, 3, \ldots$

Frequency response of Lagrange interpolators
Higher Order Integrated Wavetable Synthesis
Differentiator

- Determines aliasing and passband error
  - Ideal differentiator $H_{id}(e^{j\omega})$
    - Not realizable
  - Maximally flat design $H_{mf}(e^{j\omega})$
    - High passband roll-off
    - Reduces aliasing
  - Minimax design $H_{mm}(e^{j\omega})$
    - Wide frequency range
    - Low passband roll-off
    - Aliasing more critical
    - Used here ($\omega_c = 0.9\pi$)

| $|H(e^{j\omega})|$ |
|-----------------|
| 10              |
| 8               |
| 6               |
| 4               |
| 2               |
| 0               |

Frequency response, order $K = 2$
Higher Order Integrated Wavetable Synthesis

Differentiator

- Determines aliasing and passband error
- Ideal differentiator $H_{id}(e^{j\omega})$
  - Not realizable
- Maximally flat design $H_{mf}(e^{j\omega})$
  - High passband roll-off
  - Reduces aliasing
- Minimax design $H_{mm}(e^{j\omega})$
  - Wide frequency range
  - Low passband roll-off
  - Aliasing more critical
  - Used here ($\omega_c = 0.9\pi$)

Frequency response, order $K = 2$
Higher Order Integrated Wavetable Synthesis
Differentiator

- Determines aliasing and passband error
- Ideal differentiator $H_{id}(e^{j\omega})$
  - Not realizable
- Maximally flat design $H_{mf}(e^{j\omega})$
  - High passband roll-off
  - Reduces aliasing
- Minimax design $H_{mm}(e^{j\omega})$
  - Wide frequency range
  - Low passband roll-off
  - Aliasing more critical
  - Used here ($\omega_c = 0.9\pi$)

Frequency response, order $K = 2$
Higher Order Integrated Wavetable Synthesis
Differentiator

- Determines aliasing and passband error
- Ideal differentiator $H_{id}(e^{j\omega})$
  - Not realizable
- Maximally flat design $H_{mf}(e^{j\omega})$
  - High passband roll-off
  - Reduces aliasing
- Minimax design $H_{mm}(e^{j\omega})$
  - Wide frequency range
  - Low passband roll-off
  - Aliasing more critical
  - Used here ($\omega_c = 0.9\pi$)

Frequency response, order $K = 2$
Performance Evaluation

Increasing the order of integration

\[ K = 1, \ N = 0 \text{ [Geiger]} \]

\[ K: \text{Integration order, } N: \text{Resampling order (Lagrange interpolation)} \]
Performance Evaluation
Increasing the order of integration

$K = 1, N = 0$ [Geiger]

$K = 2, N = 0$

$K$: Integration order, $N$: Resampling order (Lagrange interpolation)
Performance Evaluation

Increasing the order of integration

\[ K = 1, \; N = 0 \] \[ K = 2, \; N = 0 \]

\[ K = 2, \; N = 1 \]

\( K \): Integration order, \( N \): Resampling order (Lagrange interpolation)
Performance Evaluation

Increasing the order of integration

\[ K = 1, \ \mathcal{N} = 0 \quad [\text{Geiger}] \]

\[ K = 2, \ \mathcal{N} = 1 \]

\[ K = 2, \ \mathcal{N} = 0 \]

\[ K = 3, \ \mathcal{N} = 1 \]

\( K \): Integration order, \( \mathcal{N} \): Resampling order (Lagrange interpolation)
Performance Evaluation
Relation Between Integration Order and Resampling Quality

\[ K = 3, \quad N = 1 \]

\( K \): Integration order, \( N \): Resampling order (Lagrange interpolation)
Performance Evaluation
Relation Between Integration Order and Resampling Quality

\[ K = 3, \ N = 1 \]

\[ K = 4, \ N = 1 \]

\( K \): Integration order, \( N \): Resampling order (Lagrange interpolation)
Performance Evaluation
Relation Between Integration Order and Resampling Quality

$K = 3, N = 1$

$K = 4, N = 1$

$K = 4, N = 3$

$K$: Integration order, $N$: Resampling order (Lagrange interpolation)
Performance Evaluation
Relation Between Integration Order and Resampling Quality

$K = 3, N = 1$

$K = 4, N = 1$

$K = 4, N = 3$

$K = 5, N = 3$

$K$: Integration order, $N$: Resampling order (Lagrange interpolation)
Performance Evaluation

Example: Sawtooth sweep

Integration order $K = 1$, resampling order $N = 0$ [Geiger]
Performance Evaluation

Example: Sawtooth sweep

Integration order $K = 5$, resampling order $N = 3$
Summary

- Anti-aliasing algorithms for wavetable synthesis
  - “Increasing the pitch”
  - Requires fewer wavetables (memory)
- Higher-order integrated wavetable synthesis
  - Provides effective anti-aliasing
  - Complexity independent of pitch change
  - General method for arbitrary signals
- Important design decisions
  - Order of integration/differentiation
  - Discrete-time Differentiator
  - Resampling method and order
  - Wavetable quantization (see paper)
Summary

- Anti-aliasing algorithms for wavetable synthesis
  - “Increasing the pitch”
  - Requires fewer wavetables (memory)
- Higher-order integrated wavetable synthesis
  - Provides effective anti-aliasing
  - Complexity independent of pitch change
  - General method for arbitrary signals
- Important design decisions
  - Order of integration/differentiation
  - Discrete-time Differentiator
  - Resampling method and order
  - Wavetable quantization (see paper)
Summary

- Anti-aliasing algorithms for wavetable synthesis
  - "Increasing the pitch"
  - Requires fewer wavetables (memory)
- Higher-order integrated wavetable synthesis
  - Provides effective anti-aliasing
  - Complexity independent of pitch change
  - General method for arbitrary signals
- Important design decisions
  - Order of integration/differentiation
  - Discrete-time Differentiator
  - Resampling method and order
  - Wavetable quantization (see paper)
Thank you for your attention!

Companion page:
http://www.idmt.fraunhofer.de/andreasfranck/dafx2012hoiws
**Additional material**

**Wavetable Quantization**

\[ K = 4, \ N = 3, \text{ quantization } b = 24 \text{ bit} \]

\[ K = 4, \ N = 1, \text{ quantization } b = 20 \text{ bit} \]

\[ K = 4, \ N = 3, \text{ quantization } b = 24 \text{ bit} \]

\[ K = 5, \ N = 3, \text{ quantization } b = 20 \text{ bit} \]

*K*: Integration order, \(N\): Resampling order (Lagrange interpolation)