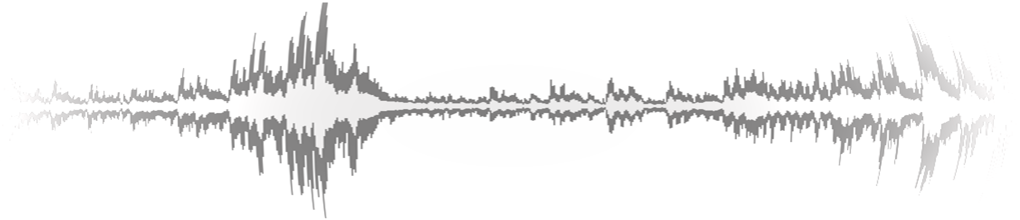
Digital Signal Processing for Music

Part 18: Sample Rate Conversion (SRC)

alexander lerch

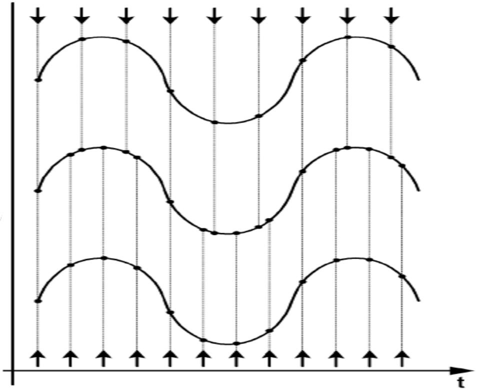


Georgia Tech Center for Music Technology
College of Design

sample rate conversion
introduction

sample rate conversion

WP:"changing the sampling rate of a discrete signal to obtain a new discrete representation of the underlying continuous signal."



sample rate conversion
introduction

sample rate conversion

WP:"changing the sampling rate of a discrete signal to obtain a new discrete representation of the underlying continuous signal."

typical applications

- audio file/media conversion
- word clock synchronization (multiple devices)
- DJing/scratching

sample rate conversion
introduction

- **terminology**
  - *synchronous*
    - clock rates are coupled
    - resampling factor stays constant
  - *asynchronous*
    - clock rates are independent
    - resampling factor may change
- **ideal result**
  - spectrum in the used band unchanged
  - spectral periodicity (determined by sample rate) changed

sample rate conversion
introduction

- **terminology**
  - *synchronous*
    - clock rates are coupled
    - resampling factor stays constant
  - *asynchronous*
    - clock rates are independent
    - resampling factor may change
- **ideal result**
  - spectrum in the used band unchanged
  - spectral periodicity (determined by sample rate) changed

sample rate conversion

upsampling by inserting zeros

- task: **upsample by integer factor L**
  1. insert $L-1$ zeros between all samples
  2. apply anti-imaging filter

sample rate conversion

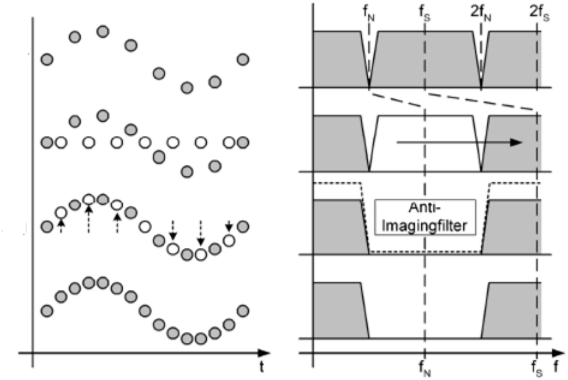
upsampling by inserting zeros

- task: **upsample by integer factor L**
  1. insert $L-1$ zeros between all samples
  2. apply anti-imaging filter

sample rate conversion

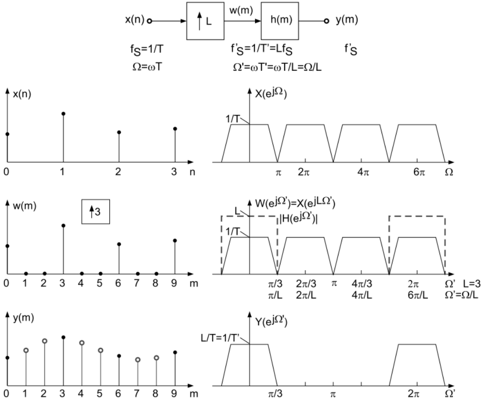
upsampling by inserting zeros

- task: **upsample by integer factor L**
  1. insert $L-1$ zeros between all samples
  2. apply anti-imaging filter



sample rate conversion
upsampling by inserting zeros

- task: **upsample by integer factor** L
  1. insert $L-1$ zeros between all samples
  2. apply anti-imaging filter



sample rate conversion

downsampling by removing samples

Georgia Tech Center for Music Technology College of Design

- task: **downsample by integer factor M**
  1. apply anti-aliasing filter
  2. take every Mth sample

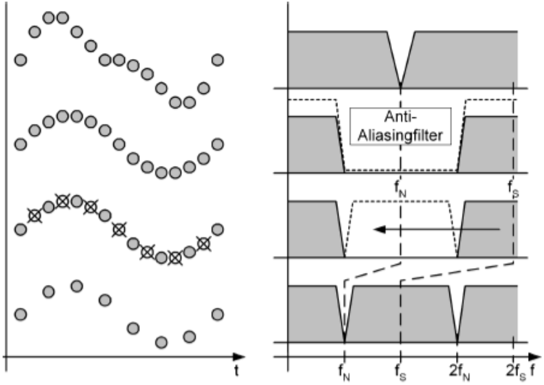
sample rate conversion
downsampling by removing samples

- task: **downsample by integer factor M**
  1. apply anti-aliasing filter
  2. take every Mth sample

sample rate conversion

downsampling by removing samples

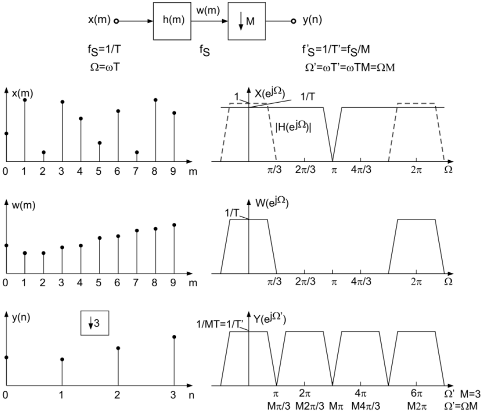
- task: **downsample by integer factor M**
  1. apply anti-aliasing filter
  2. take every Mth sample



sample rate conversion

downsampling by removing samples

- task: **downsample by integer factor** M
  1. apply anti-aliasing filter
  2. take every Mth sample



sample rate conversion

resampling by rational factor

- task: **convert sample rate to any** other (coupled) sample rate
  1. convert sample rate ratio to integer factors
     e.g.: $^{48}/_{44.1} \Rightarrow L = 160, M = 147$
  2. insert zeros
  3. apply anti-imaging filter
  4. apply anti-aliasing filter
  5. remove samples

sample rate conversion

resampling by rational factor

Georgia Tech Center for Music Technology College of Design

- task: **convert sample rate to any** other (coupled) sample rate
  1. convert sample rate ratio to integer factors
     e.g.: $^{48}/_{44.1} \Rightarrow L = 160, M = 147$
  2. insert zeros
  3. apply anti-imaging filter
  4. apply anti-aliasing filter
  5. remove samples

sample rate conversion

resampling by rational factor

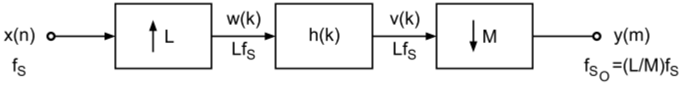
Georgia Tech Center for Music Technology College of Design

- task: **convert sample rate to any** other (coupled) sample rate
  1. convert sample rate ratio to integer factors
     e.g.: $^{48}/_{44.1} \Rightarrow L = 160, M = 147$
  2. insert zeros
  3. apply anti-imaging filter
  4. apply anti-aliasing filter
  5. remove samples

sample rate conversion

resampling by rational factor

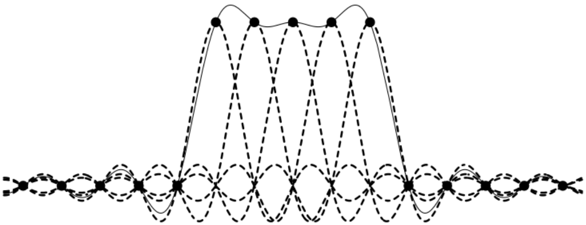
- task: **convert sample rate to any** other (coupled) sample rate
  1. convert sample rate ratio to integer factors
     e.g.: $^{48}/_{44.1} \Rightarrow L = 160, M = 147$
  2. insert zeros
  3. apply anti-imaging filter
  4. apply anti-aliasing filter
  5. remove samples



sample rate conversion
sinc interpolation 1/2

- perfect reconstruction of the sampled spectrum is possible with ideal filter

⇒ resampling should be possible by time domain convolution with sinc



$$x(i-\alpha) = \sum_{m=-\infty}^{\infty} x(m) \frac{\Omega_C}{\pi} \frac{sin\left(\Omega_C(i-\alpha-m)\right)}{\Omega_C(i-\alpha-m)}$$

Ω_C is the cutoff frequency of the ideal lowpass

sample rate conversion

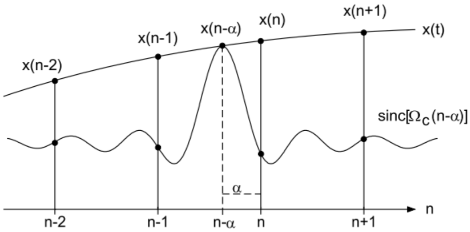
sinc interpolation 1/2

- perfect reconstruction of the sampled spectrum is possible with ideal filter

$\Rightarrow$ resampling should be possible by time domain convolution with sinc

$$x(i-\alpha) = \sum_{m=-\infty}^{\infty} x(m)\frac{\Omega_C}{\pi}\frac{\sin\left(\Omega_C(i-\alpha-m)\right)}{\Omega_C(i-\alpha-m)}$$

Ω_C is the cutoff frequency of the ideal lowpass



sample rate conversion
sinc interpolation 2/2

- practical implementation: **windowed sinc**

sample rate conversion
polynomial interpolation

- **interpolation methods**
  - can be interpreted as filters with time-variant filter coefficients
  - not based on traditional filter desing methods
- polynomial interpolation

$$f(t) = \sum_{k=0}^{\mathcal{O}} x_k p_k(t)$$

$$p_k(t) = \prod_{j=0}^{\mathcal{O}} \frac{t - t_j}{t_k - t_j}$$

sample rate conversion
polynomial interpolation

- **interpolation methods**
  - can be interpreted as filters with time-variant filter coefficients
  - not based on traditional filter desing methods
- polynomial interpolation

$$
\begin{aligned}
f(t) &= \sum_{k=0}^{\mathcal{O}} x_k p_k(t) \\
p_k(t) &= \prod_{j=0}^{\mathcal{O}} \frac{t - t_j}{t_k - t_j}
\end{aligned}
$$

sample rate conversion

polynomial interpolation example

Georgia Tech Center for Music Technology College of Design

$$
\begin{aligned}
x(t) &= \frac{1}{t} \\
\text{nodes: } t &= [2, 4, 5] \\
p_0(t) &= \frac{(t-4)(t-5)}{(2-4)(2-5)} = \frac{(t-4)(t-5)}{6} \\
p_1(t) &= \frac{(t-2)(t-5)}{(4-2)(4-5)} = -\frac{(t-2)(t-5)}{2} \\
p_2(t) &= \frac{(t-2)(t-4)}{(5-2)(5-4)} = \frac{(t-2)(t-4)}{3} \\
f(t) &= \sum_{k=0}^{\mathcal{O}} x_k p_k(t) \\
\Rightarrow f(t) &= p_0\frac{1}{2} + p_1\frac{1}{4} + p_2\frac{1}{5} \\
&= 0.025t^2 - 0.275t + 0.95.
\end{aligned}
$$

sample rate conversion

polynomial interpolation example

$$
\begin{aligned}
x(t) &= \frac{1}{t} \\
\text{nodes: } t &= [2, 4, 5] \\
p_0(t) &= \frac{(t-4)(t-5)}{(2-4)(2-5)} = \frac{(t-4)(t-5)}{6} \\
p_1(t) &= \frac{(t-2)(t-5)}{(4-2)(4-5)} = -\frac{(t-2)(t-5)}{2} \\
p_2(t) &= \frac{(t-2)(t-4)}{(5-2)(5-4)} = \frac{(t-2)(t-4)}{3} \\
f(t) &= \sum_{k=0}^{\mathcal{O}} x_k p_k(t) \\
\Rightarrow f(t) &= p_0 \frac{1}{2} + p_1 \frac{1}{4} + p_2 \frac{1}{5} \\
&= 0.025t^2 - 0.275t + 0.95.
\end{aligned}
$$

sample rate conversion

polynomial interpolation example

$$
\begin{aligned}
x(t) &= \frac{1}{t} \\
\text{nodes: } t &= [2, 4, 5] \\
p_0(t) &= \frac{(t-4)(t-5)}{(2-4)(2-5)} = \frac{(t-4)(t-5)}{6} \\
p_1(t) &= \frac{(t-2)(t-5)}{(4-2)(4-5)} = -\frac{(t-2)(t-5)}{2} \\
p_2(t) &= \frac{(t-2)(t-4)}{(5-2)(5-4)} = \frac{(t-2)(t-4)}{3} \\
f(t) &= \sum_{k=0}^{\mathcal{O}} x_k p_k(t) \\
\Rightarrow f(t) &= p_0 \frac{1}{2} + p_1 \frac{1}{4} + p_2 \frac{1}{5} \\
&= 0.025t^2 - 0.275t + 0.95.
\end{aligned}
$$

sample rate conversion

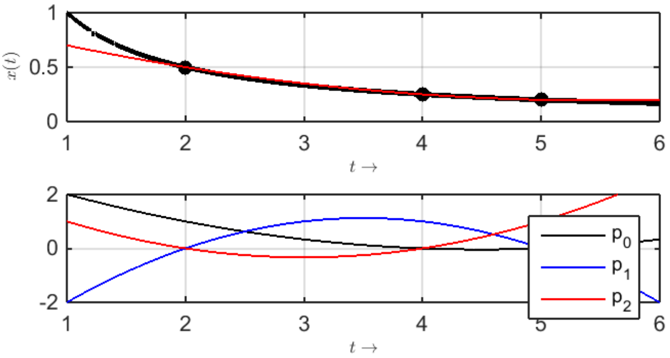
polynomial interpolation example

$$
\begin{aligned}
x(t) &= \frac{1}{t} \\
\text{nodes: } t &= [2, 4, 5] \\
p_0(t) &= \frac{(t-4)(t-5)}{(2-4)(2-5)} = \frac{(t-4)(t-5)}{6} \\
p_1(t) &= \frac{(t-2)(t-5)}{(4-2)(4-5)} = -\frac{(t-2)(t-5)}{2} \\
p_2(t) &= \frac{(t-2)(t-4)}{(5-2)(5-4)} = \frac{(t-2)(t-4)}{3} \\
f(t) &= \sum_{k=0}^{\mathcal{O}} x_k p_k(t) \\
\Rightarrow f(t) &= p_0 \frac{1}{2} + p_1 \frac{1}{4} + p_2 \frac{1}{5} \\
&= 0.025t^2 - 0.275t + 0.95.
\end{aligned}
$$

sample rate conversion

polynomial interpolation example

Georgia Tech Center for Music Technology College of Design



sample rate conversion

special case: linear interpolation

- 1st order $\rightarrow$ 2 points

$$
\begin{aligned}
x(t) &= \frac{1}{t} \\
\text{nodes: } t &= [2, 4] \\
p_0(t) &= \frac{(t-4)}{(2-4)} = \frac{(4-t)}{2} \\
p_1(t) &= \frac{(t-2)}{(4-2)} = \frac{(t-2)}{2} \\
f(t) &= \sum_{k=0}^{\mathcal{O}} x_k p_k(t) \\
\Rightarrow f(t) &= p_0 \frac{1}{2} + p_1 \frac{1}{4} \\
&= -\frac{1}{8} t + \frac{3}{4}.
\end{aligned}
$$

sample rate conversion

special case: linear interpolation

- 1st order $\rightarrow$ 2 points

$$
\begin{aligned}
x(t) &= \frac{1}{t} \\
\text{nodes: } t &= [2, 4] \\
p_0(t) &= \frac{(t-4)}{(2-4)} = \frac{(4-t)}{2} \\
p_1(t) &= \frac{(t-2)}{(4-2)} = \frac{(t-2)}{2} \\
f(t) &= \sum_{k=0}^{\mathcal{O}} x_k p_k(t) \\
\Rightarrow f(t) &= p_0 \frac{1}{2} + p_1 \frac{1}{4} \\
&= -\frac{1}{8} t + \frac{3}{4}.
\end{aligned}
$$

sample rate conversion
special case: linear interpolation

- 1st order $\rightarrow$ 2 points

$$
\begin{aligned}
x(t) &= \frac{1}{t} \\
\text{nodes: } t &= [2, 4] \\
p_0(t) &= \frac{(t-4)}{(2-4)} = \frac{(4-t)}{2} \\
p_1(t) &= \frac{(t-2)}{(4-2)} = \frac{(t-2)}{2} \\
f(t) &= \sum_{k=0}^{\mathcal{O}} x_k p_k(t) \\
\Rightarrow f(t) &= p_0 \frac{1}{2} + p_1 \frac{1}{4} \\
&= -\frac{1}{8} t + \frac{3}{4}.
\end{aligned}
$$

sample rate conversion
special case: linear interpolation

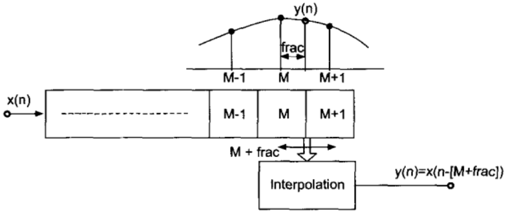
Georgia Tech Center for Music Technology College of Design

- 1st order $\rightarrow$ 2 points

$$
\begin{aligned}
x(t) &= \frac{1}{t} \\
\text{nodes: } t &= [2, 4] \\
p_0(t) &= \frac{(t-4)}{(2-4)} = \frac{(4-t)}{2} \\
p_1(t) &= \frac{(t-2)}{(4-2)} = \frac{(t-2)}{2} \\
f(t) &= \sum_{k=0}^{\mathcal{O}} x_k p_k(t) \\
\Rightarrow f(t) &= p_0 \frac{1}{2} + p_1 \frac{1}{4} \\
&= -\frac{1}{8} t + \frac{3}{4}.
\end{aligned}
$$

sample rate conversion

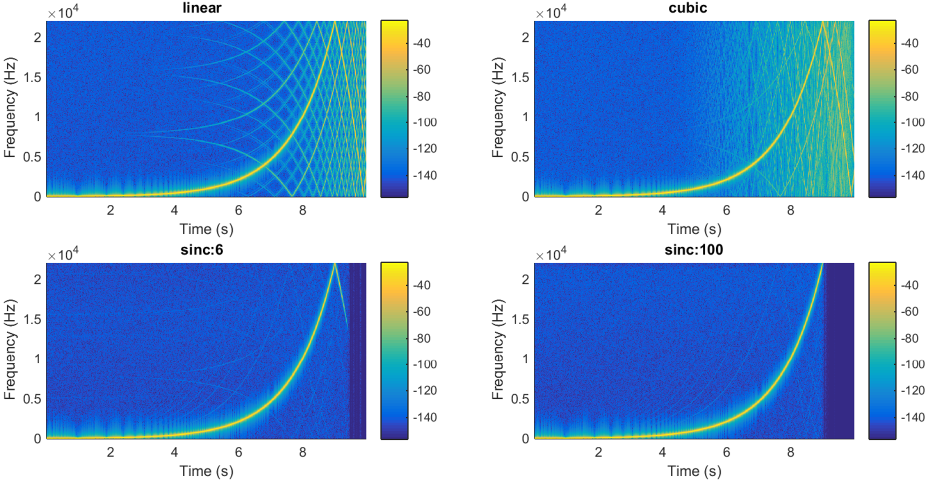
special case: linear interpolation



$$\hat{x} = x_l \cdot (1 - frac) + x_r \cdot frac$$

sample rate conversion

interpolation comparison



sample rate conversion

downsampling audio examples

	orig (48 kHz)	ds (6 kHz, w/o filt)	ds (6 kHz, w/ filt)
sax			
big band			

sample rate conversion

summary

- resampling: estimate different sample points of underlying continuous signal
- as with sampling, proper filtering has to take place
- some interpolation approaches have filter “built-in”
- perfect reconstruction impossible (infinite sinc), however, perceptually artifact-free resampling is possible
  - main issue: filter cut-off and steepness vs. aliasing