Digital Audio Synthesis and Effects based on Physical Models

Julius Smith
CCRMA, Stanford University

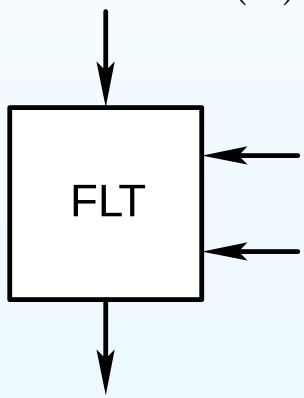
DAFx-2006 Keynote II

September 19, 2006

Early Digital Audio Effects

First Digital Audio Effect?

The `FLT` *unit generator* in Max Mathews’ `Music V` program (begun in the 1950s) — a *digital two-pole resonator*:



$$R = e^{-\pi B/f_s}$$
$$\theta = f_c/f_s$$

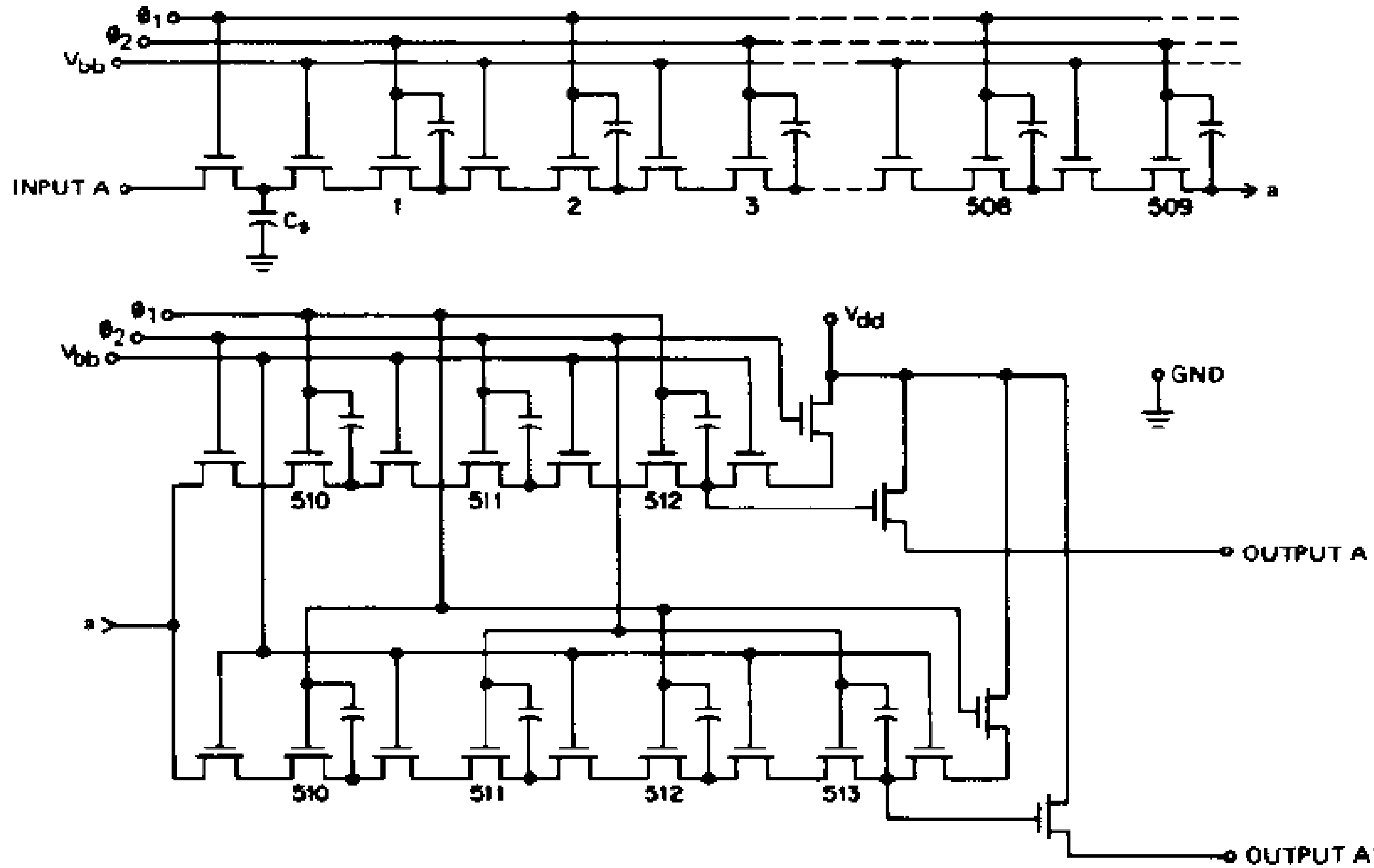
$$\begin{aligned} y(n) = x(n) & \\ & -[2R\cos(\theta)]y(n-1) \\ & +R^2 y(n-2) \end{aligned}$$

$$H(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

- Written in Fortran
- Nonrealtime

First Discrete-Time Effects Chip?

Reticon Bucket Brigade Delay Line (discrete time, analog amplitude):

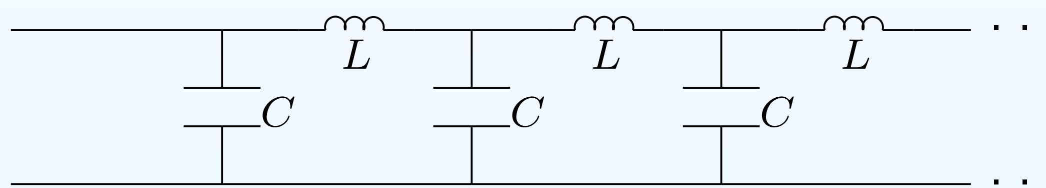


(from 1977 Reticon SAD-1024 Dual Analog Delay Line Datasheet)

What did we do before digital?

Analog delay techniques

- Magnetic tape loop
- Surface acoustic wave devices
- LC ladder (analog delay line):



  - Lumped model of an electric *transmission line*
  - Used in early electronic vocal tract models (1940s)
  - Used in Hammond organ “Scanner Vibrato”

Early DAFx
- First DAFt?
- Bucket Brigade DL
- Analog Delay
- Scanner Vibrato
- Sound Examples
- KL Music
- “Daisy”
- “Shiela”

Delay Effects

Waveguide Models

Commuted Synthesis

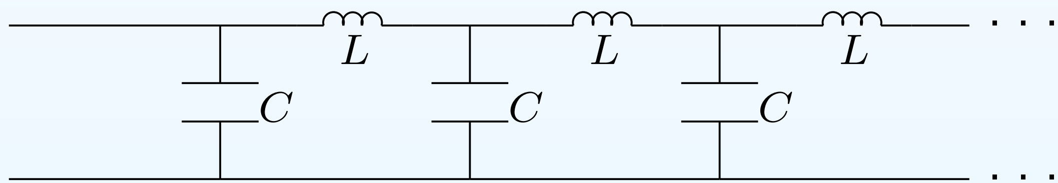
Summary

Related Topics

What did we do before digital?

Analog delay techniques

- Magnetic tape loop
- Surface acoustic wave devices
- LC ladder (analog delay line):

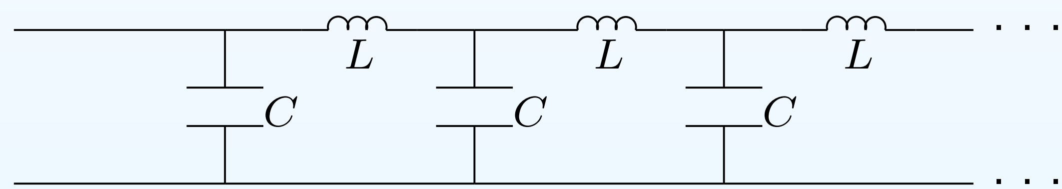


  - Lumped model of an electric *transmission line*
  - Used in early electronic vocal tract models (1940s)
  - Used in Hammond organ “Scanner Vibrato”

What did we do before digital?

Analog delay techniques

- Magnetic tape loop
- Surface acoustic wave devices
- LC ladder (analog delay line):

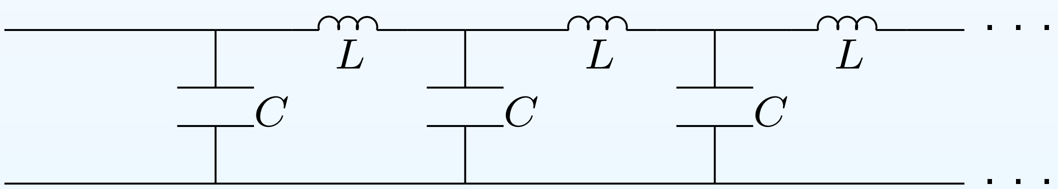


  - Lumped model of an electric *transmission line*
  - Used in early electronic vocal tract models (1940s)
  - Used in Hammond organ “Scanner Vibrato”

What did we do before digital?

Analog delay techniques

- Magnetic tape loop
- Surface acoustic wave devices
- LC ladder (analog delay line):

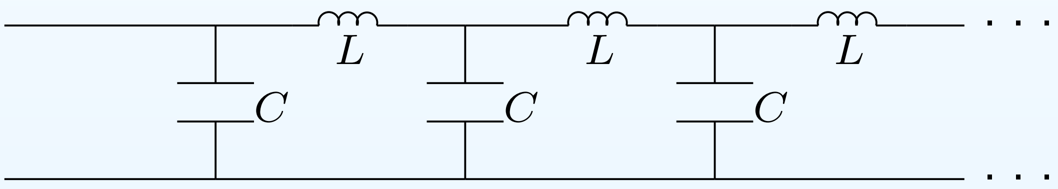


  - Lumped model of an electric *transmission line*
  - Used in early electronic vocal tract models (1940s)
  - Used in Hammond organ “Scanner Vibrato”

What did we do before digital?

Analog delay techniques

- Magnetic tape loop
- Surface acoustic wave devices
- LC ladder (analog delay line):

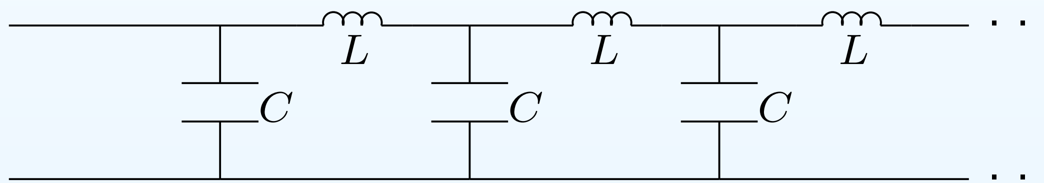


- Lumped model of an electric *transmission line*
- Used in early electronic vocal tract models (1940s)
- Used in Hammond organ “Scanner Vibrato”

What did we do before digital?

Analog delay techniques

- Magnetic tape loop
- Surface acoustic wave devices
- LC ladder (analog delay line):



  - Lumped model of an electric *transmission line*
  - Used in early electronic vocal tract models (1940s)
  - Used in Hammond organ “Scanner Vibrato”

What did we do before digital?

Early DAFx
- First DAFt?
- Bucket Brigade DL
- Analog Delay
- Scanner Vibrato
- Sound Examples
- KL Music
- “Daisy”
- “Shiela”

Delay Effects

Waveguide Models

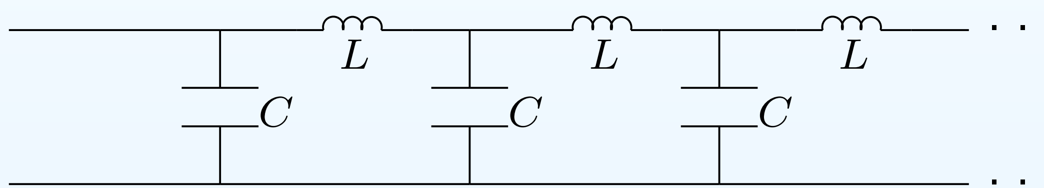
Commuted Synthesis

Summary

Related Topics

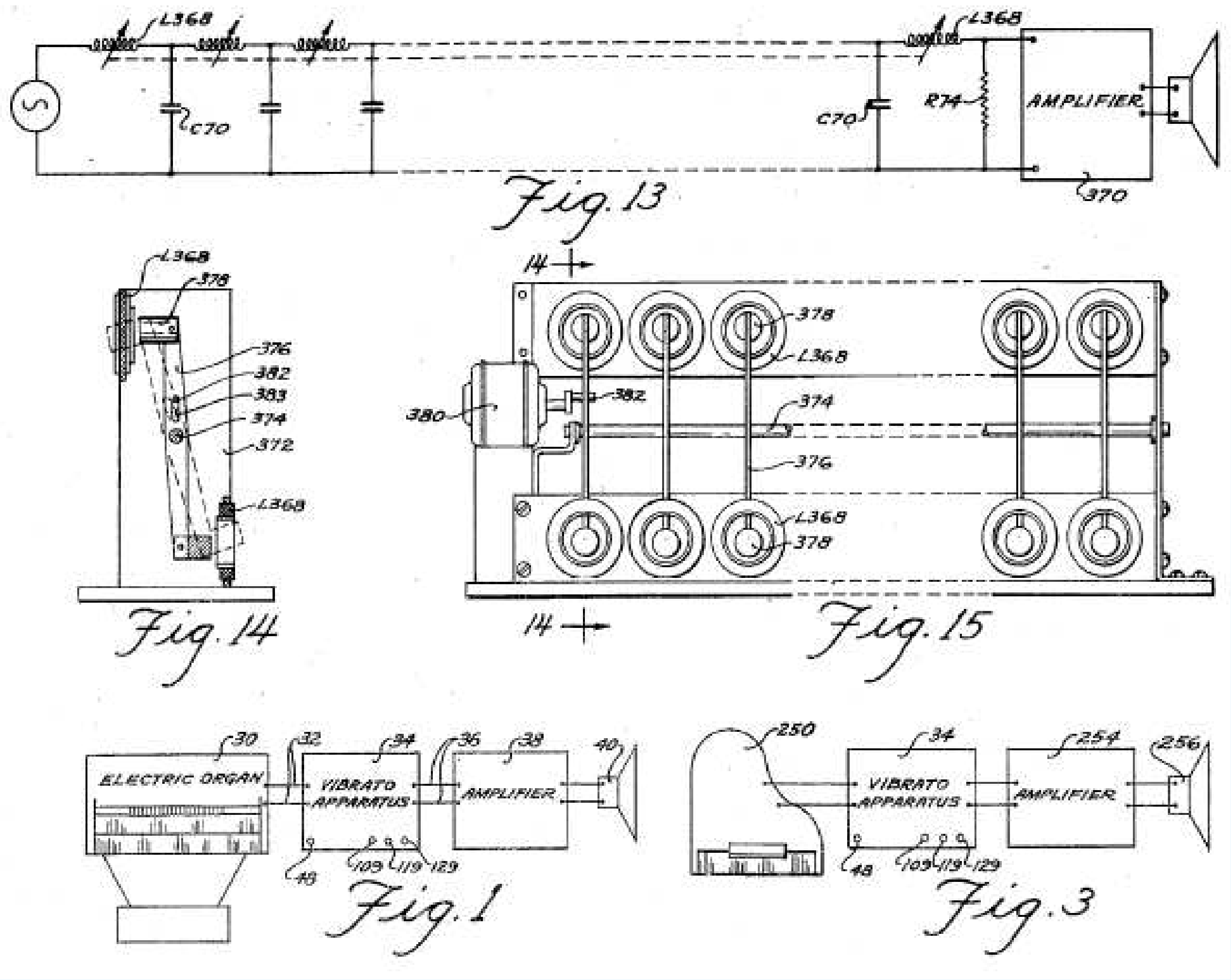
Analog delay techniques

- Magnetic tape loop
- Surface acoustic wave devices
- LC ladder (analog delay line):



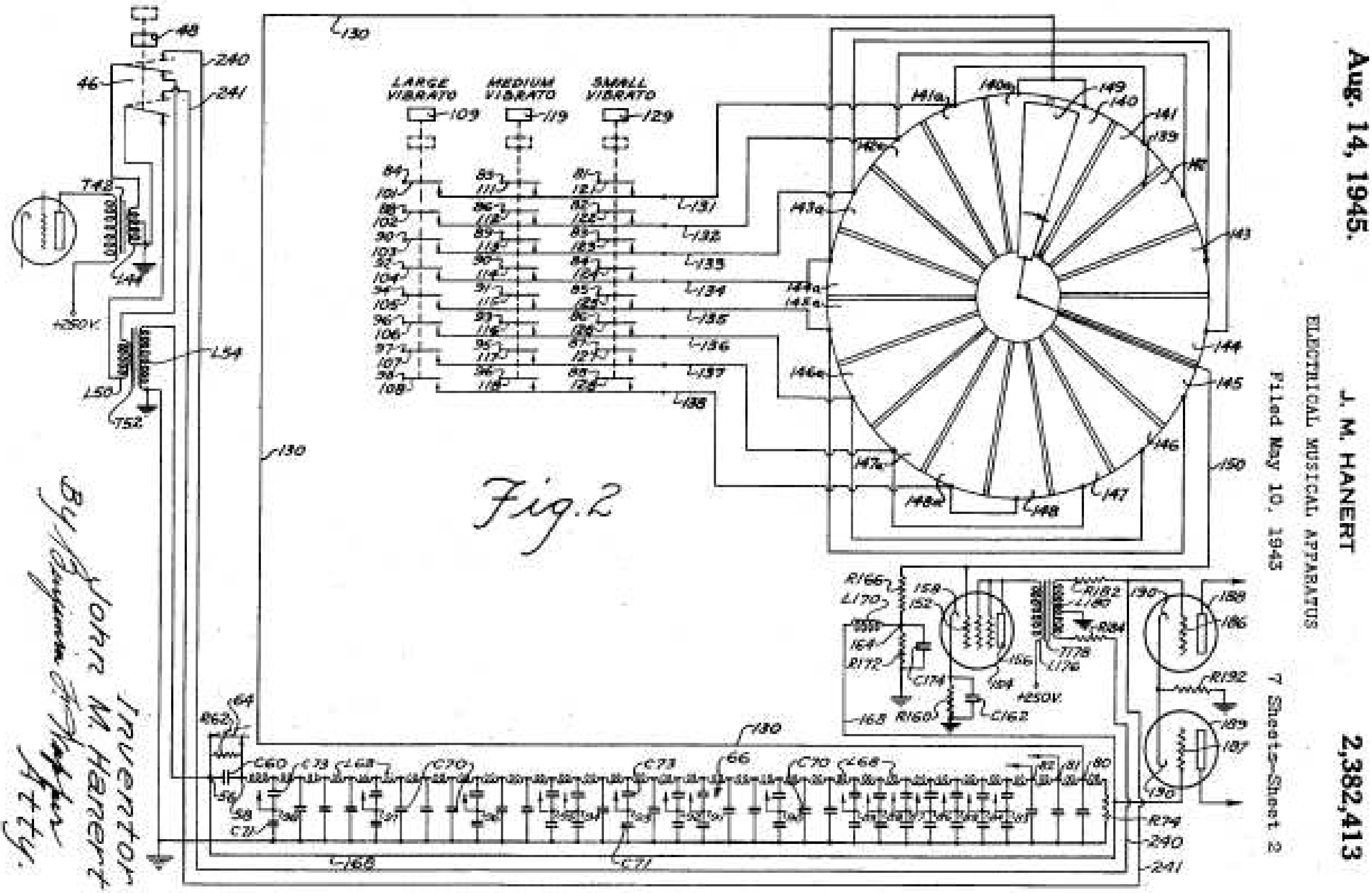
  - Lumped model of an electric *transmission line*
  - Used in early electronic vocal tract models (1940s)
  - Used in Hammond organ “Scanner Vibrato”

Scanner Vibrato



John Hanert's Scanner Vibrato patent for the Hammond Organ (early 1940s).

Hammond Scanner Vibrato



Hammond Scanner Vibrato Sound Examples

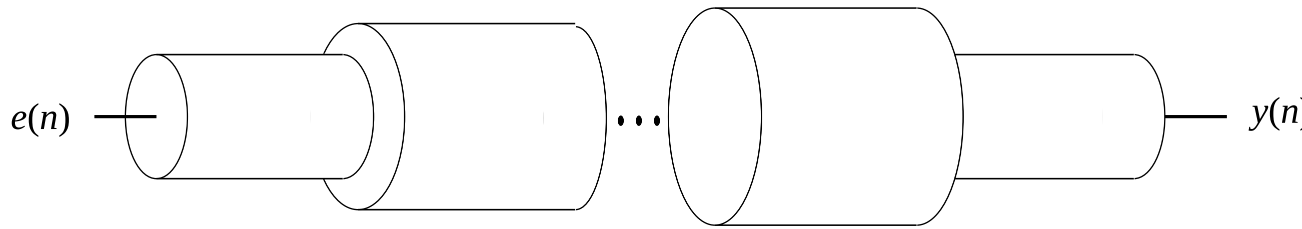
- “Dry” Organ

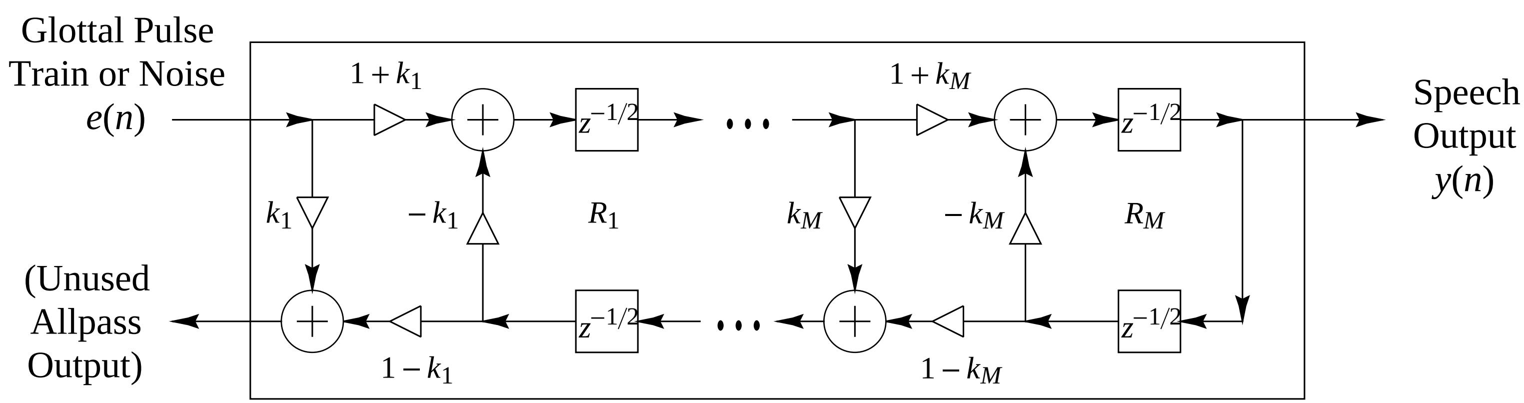
- Vibrato 1 Mode
- Vibrato 2 Mode
- Vibrato 3 Mode

- Chorus 1 Mode
- Chorus 2 Mode
- Chorus 3 Mode

- Source: Juergen Haible’s home page:
  `http://jhaible.heim.at/scanner_vibrato/jh_scanner_vibrato.html`

Kelly-Lochbaum Vocal Tract Model





Kelly-Lochbaum Vocal Tract Model (Piecewise Cylindrical)

John L. Kelly and Carol Lochbaum (1962)

Sound Example

“Bicycle Built for Two”: (WAV) (MP3)

- Vocal part by Kelly and Lochbaum (1961)

Sound Example

“Bicycle Built for Two”: (WAV) (MP3)

- Vocal part by Kelly and Lochbaum (1961)
- Musical accompaniment by Max Mathews

Sound Example

“Bicycle Built for Two”: (WAV) (MP3)

- Vocal part by Kelly and Lochbaum (1961)
- Musical accompaniment by Max Mathews
- Computed on an IBM 704

Sound Example

“Bicycle Built for Two”: (WAV) (MP3)

- Vocal part by Kelly and Lochbaum (1961)
- Musical accompaniment by Max Mathews
- Computed on an IBM 704
- Based on Russian speech-vowel data from Gunnar Fant’s book

Sound Example

“Bicycle Built for Two”: (WAV) (MP3)

- Vocal part by Kelly and Lochbaum (1961)
- Musical accompaniment by Max Mathews
- Computed on an IBM 704
- Based on Russian speech-vowel data from Gunnar Fant’s book
- Probably the first digital physical-modeling synthesis sound example by any method

Sound Example

“Bicycle Built for Two”: (WAV) (MP3)

- Vocal part by Kelly and Lochbaum (1961)
- Musical accompaniment by Max Mathews
- Computed on an IBM 704
- Based on Russian speech-vowel data from Gunnar Fant’s book
- Probably the first digital physical-modeling synthesis sound example by any method
- Inspired Arthur C. Clarke to adapt it for “2001: A Space Odyssey” — the computer’s “first song”

Early DAFx
- First DAFt?
- Bucket Brigade DL
- Analog Delay
- Scanner Vibrato
- Sound Examples
- KL Music
- “Daisy”
- “Shiela”

Delay Effects

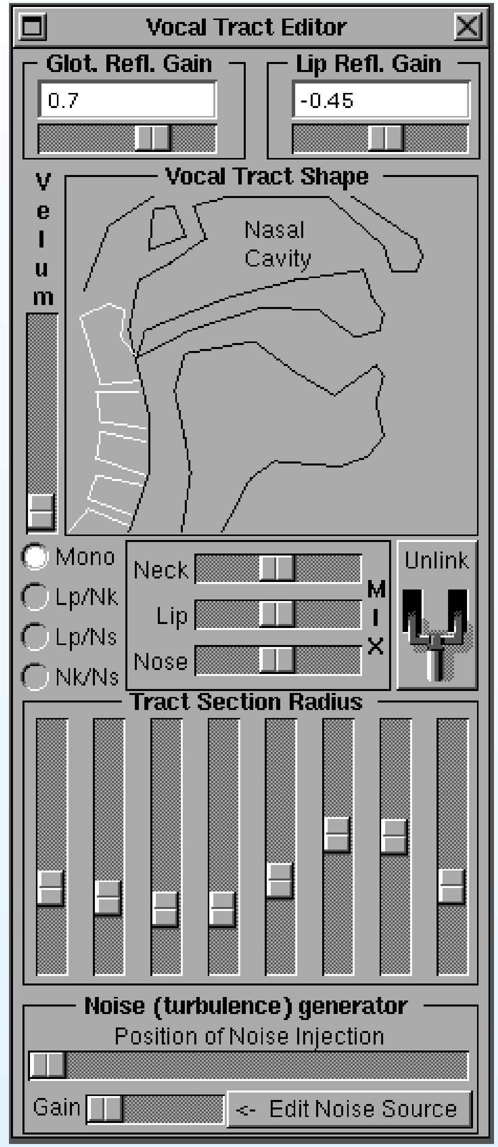
Waveguide Models

Commuted Synthesis

Summary

Related Topics

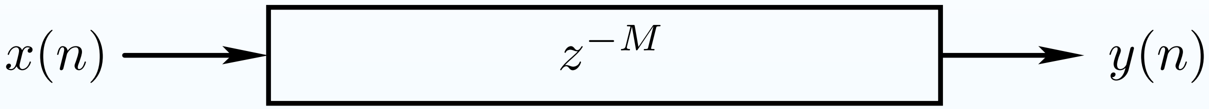
“Shiela” Sound Examples by Perry Cook (1990)



- Diphones: (WAV) (MP3)
- Nasals: (WAV) (MP3)
- Scales: (WAV) (MP3)
- “Shiela”: (WAV) (MP3)

The Versatile Delay Line

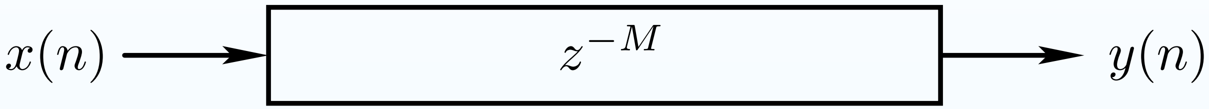
The Delay Line



- $y(n) = x(n-M),\, n = 0, 1, 2, \ldots$
- $x(-1) = x(-2) = \cdots = x(-M) \stackrel{\Delta}{=} 0$

- Models *plane wave* propagation in one direction
- Models traveling waves on an *ideal string* in one direction

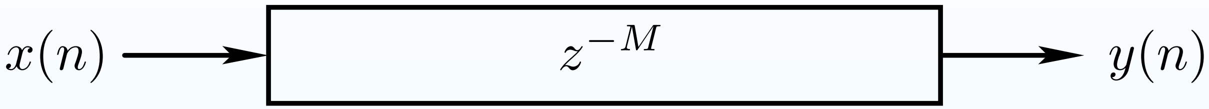
The Delay Line



- $y(n) = x(n-M), n = 0, 1, 2, \dots$
- $x(-1) = x(-2) = \cdots = x(-M) \stackrel{\Delta}{=} 0$

- Models *plane wave* propagation in one direction
- Models traveling waves on an *ideal string* in one direction

The Delay Line



- $y(n) = x(n-M), n = 0, 1, 2, \ldots$
- $x(-1) = x(-2) = \cdots = x(-M) \triangleq 0$

- Models *plane wave* propagation in one direction
- Models traveling waves on an *ideal string* in one direction

Delay Line in C


```
static double D[M]; /* initialized to zero */
static long ptr=0;  /* read-write offset */

double delayline(double x)
{
  double y = D[ptr]; /* read operation */
  D[ptr++] = x;      /* write operation */
  if (ptr >= M) { ptr -= M; } /* wrap ptr */
  return y;
}
```

Circular buffer in software

Lossy and Dispersive Traveling Waves

In all acoustic systems of interest, propagation losses *vary with frequency*.

- For *passive* media,

$$\left|H(e^{j\omega T})\right| \leq 1, \; \forall\omega$$

- For lossless, dispersive wave propagation, the filter is "allpass," i.e.,

$$\left|H(e^{j\omega T})\right| \equiv 1, \; \forall\omega$$

- Filter is *linear and time-invariant (LTI)* when the medium is described by *constant-coefficient linear differential equations.*

Lossy and Dispersive Traveling Waves

In all acoustic systems of interest, propagation losses *vary with frequency*.

- For *passive* media,

$$\left|H(e^{j\omega T})\right| \leq 1, \ \forall\omega$$

- For lossless, dispersive wave propagation, the filter is “allpass,” i.e.,

$$\left|H(e^{j\omega T})\right| \equiv 1, \ \forall\omega$$

- Filter is *linear and time-invariant (LTI)* when the medium is described by *constant-coefficient linear differential equations.*

Lossy and Dispersive Traveling Waves

In all acoustic systems of interest, propagation losses *vary with frequency*.

- For *passive* media,

$$\left|H(e^{j\omega T})\right| \leq 1, \ \forall\omega$$

- For lossless, dispersive wave propagation, the filter is "allpass," i.e.,

$$\left|H(e^{j\omega T})\right| \equiv 1, \ \forall\omega$$

- Filter is *linear and time-invariant (LTI)* when the medium is described by *constant-coefficient linear differential equations.*

Lossy and Dispersive Traveling Waves

In all acoustic systems of interest, propagation losses *vary with frequency*.

- For *passive* media,

$$\left|H(e^{j\omega T})\right| \leq 1, \ \forall\omega$$

- For lossless, dispersive wave propagation, the filter is “allpass,” i.e.,

$$\left|H(e^{j\omega T})\right| \equiv 1, \ \forall\omega$$

- Filter is *linear and time-invariant (LTI)* when the medium is described by *constant-coefficient linear differential equations.*

Air Absorption

From Moorer 1979 (“About this Reverberation Business”):

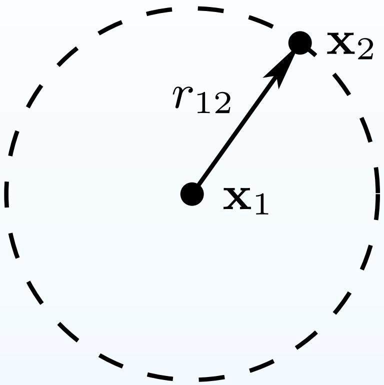
Intensity of a plane wave r meters from a vibrating-plane source

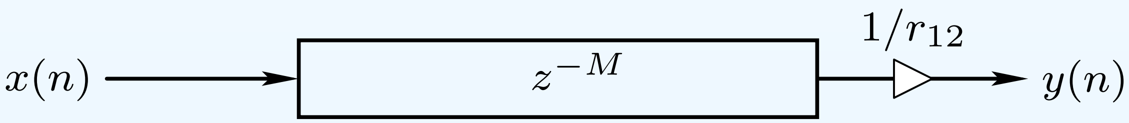
$$I(r) = I_0 e^{-r/\tau_r}$$

Relative Humidity	Frequency in Hz 1000	2000	3000	4000
40	5.6	16	30	105
50	5.6	12	26	90
60	5.6	12	24	73
70	5.6	12	22	63

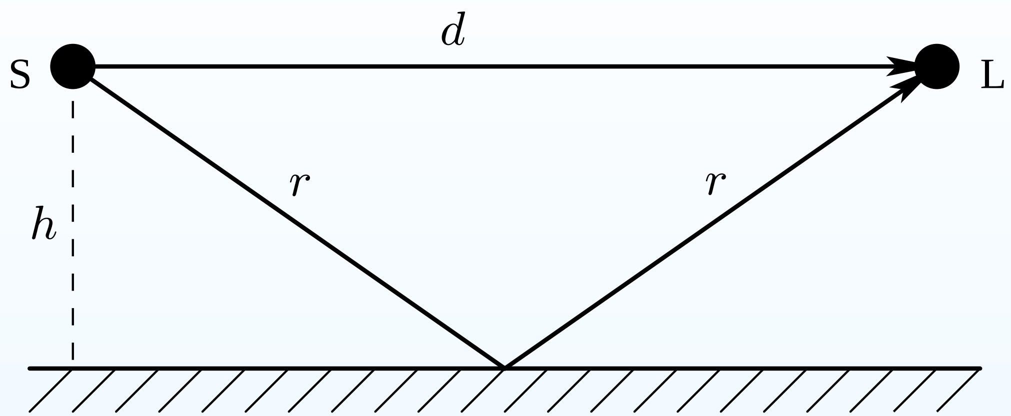
Attenuation in dB per kilometer at STP

Spreading Loss

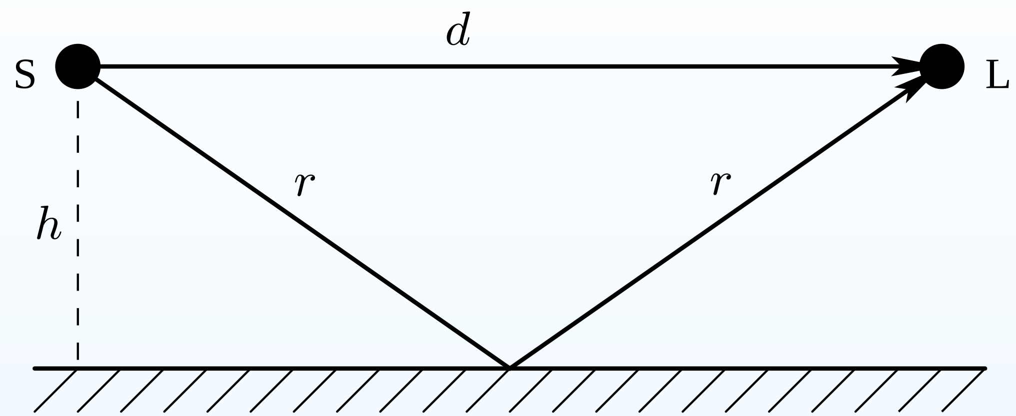


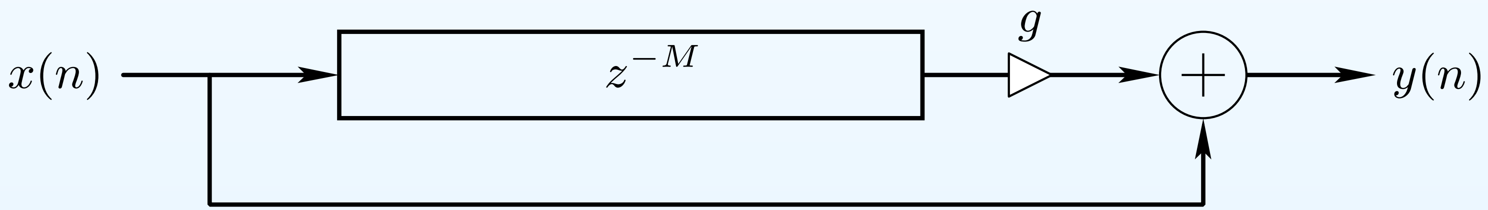


Acoustic Echo

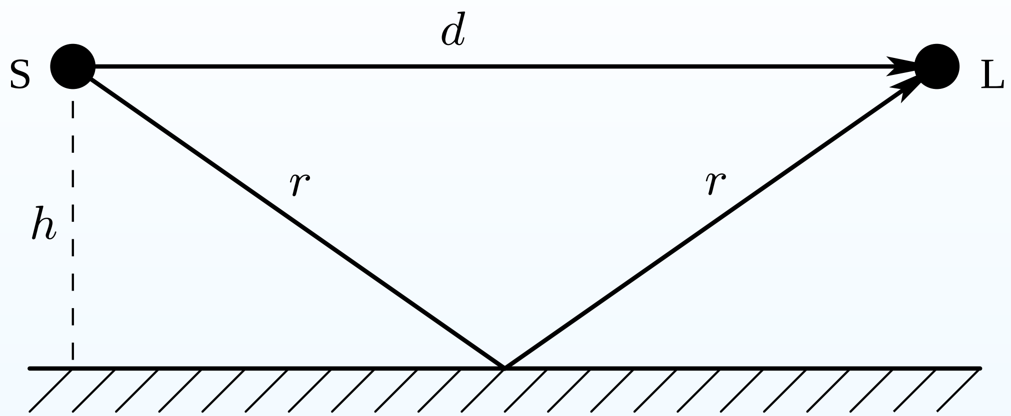


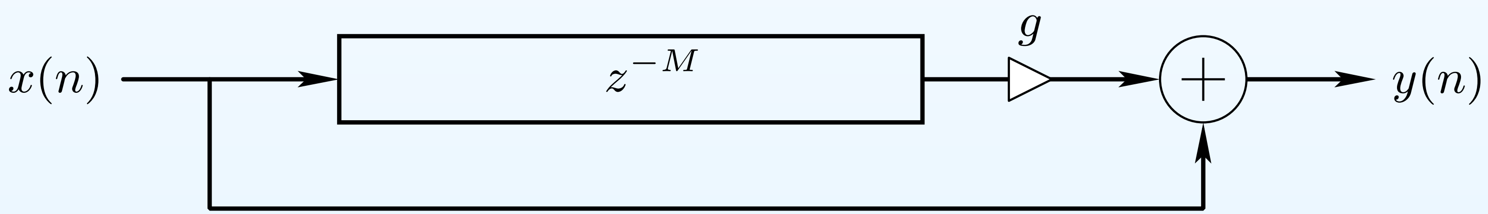
Acoustic Echo





Acoustic Echo

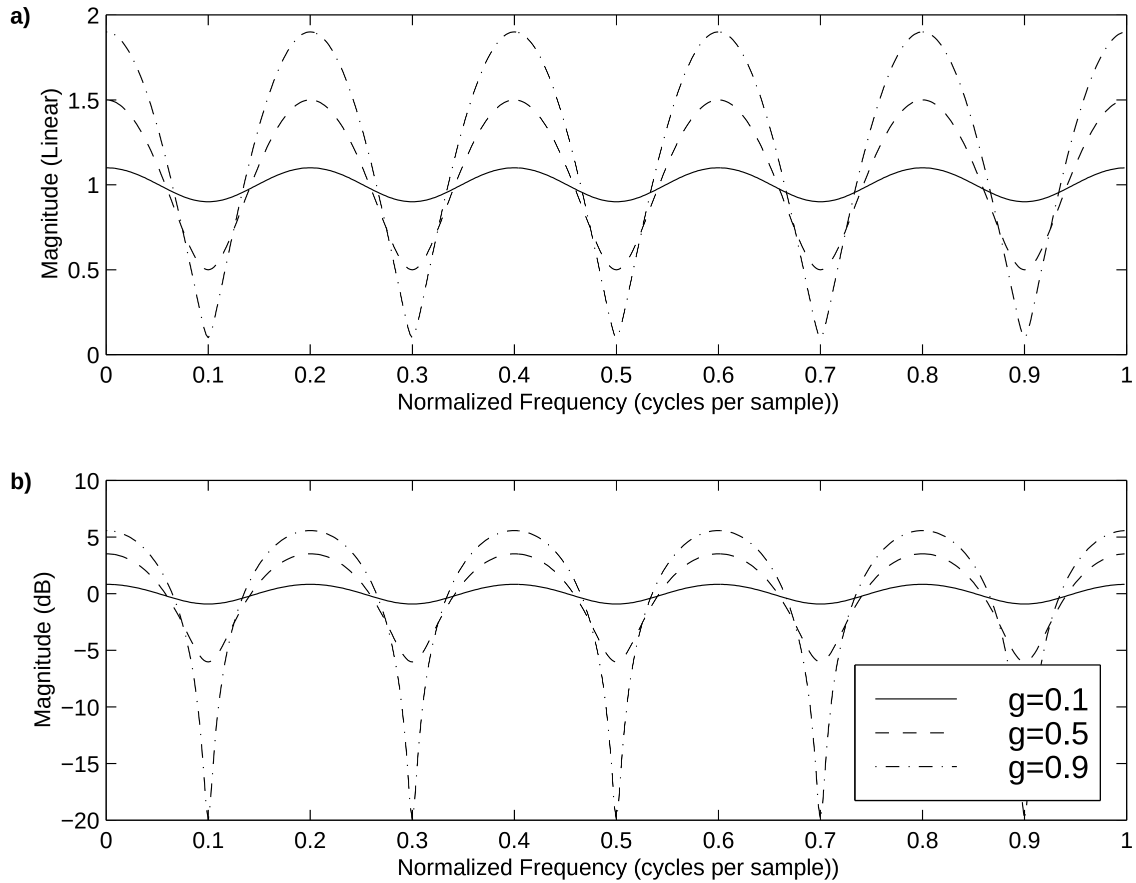




$$M = \frac{2r - d}{cT}$$

$$g = \frac{1/2r}{1/d} = \frac{1}{\sqrt{1 + (2h/d)^2}}$$

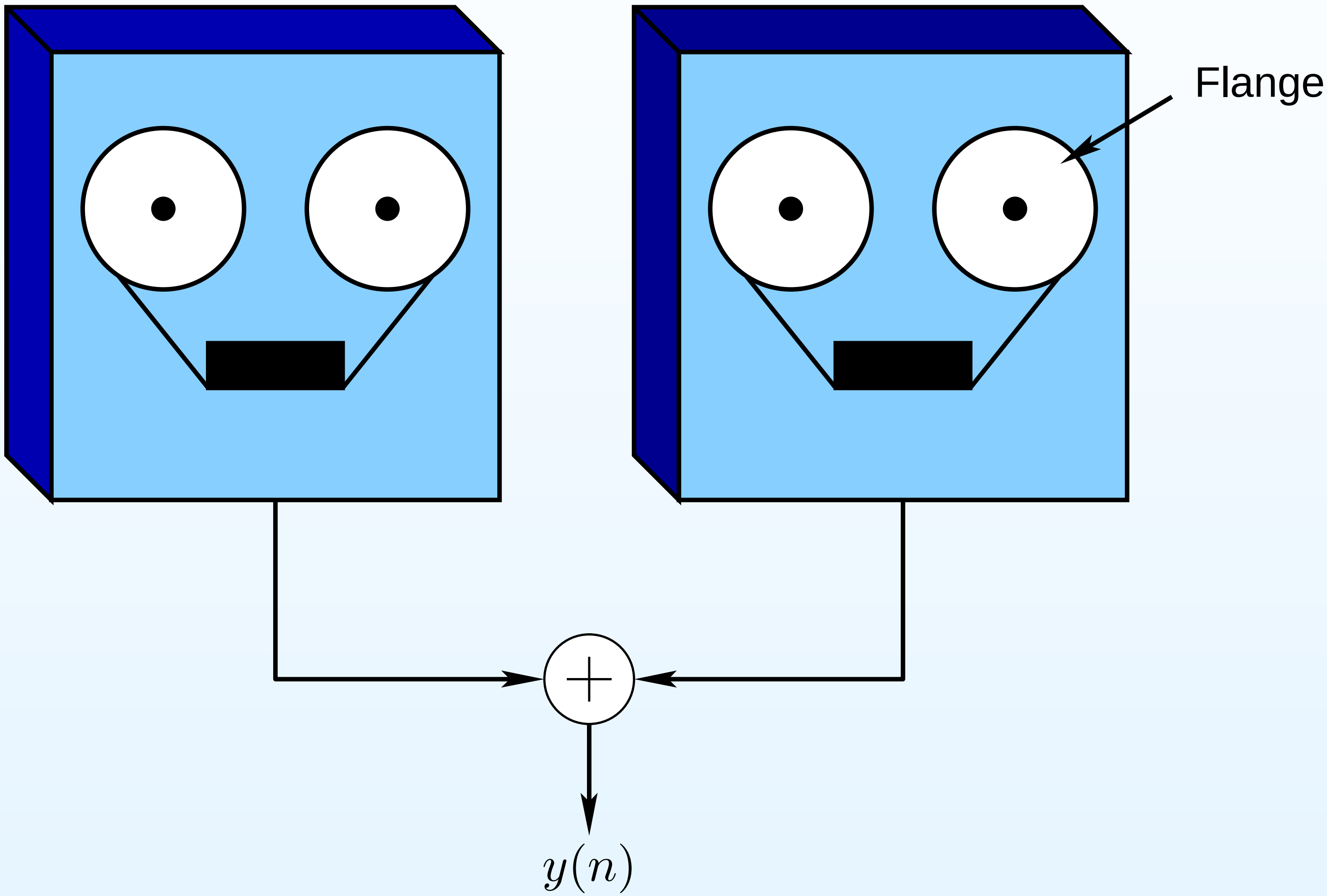
Feed-Forward Comb-Filter Amp Response



Linear (top) and decibel (bottom) amplitude scales

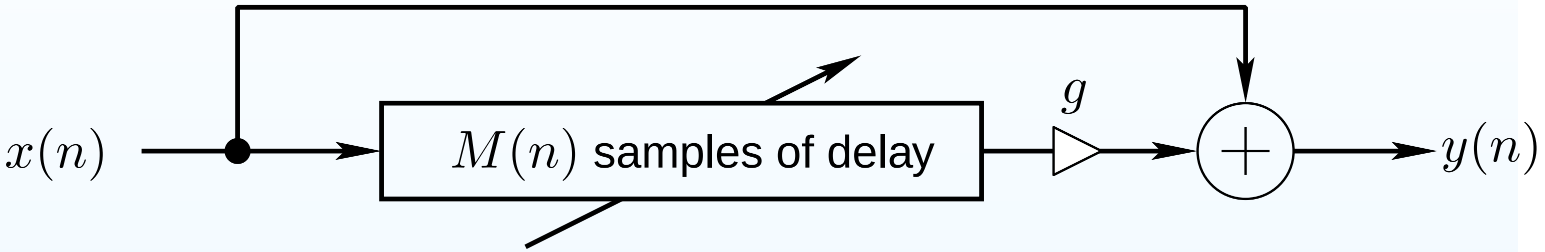
- $H(z) = 1 + gz^{-M}, \quad M = 5, \quad g = 0.1, 0.5, 0.9$
- $G(\omega) \stackrel{\Delta}{=} \left|H(e^{j\omega T})\right| = \left|1 + ge^{-jM\omega T}\right|$
- In *flangers*, these nulls move slowly with time.

Flanging Effect



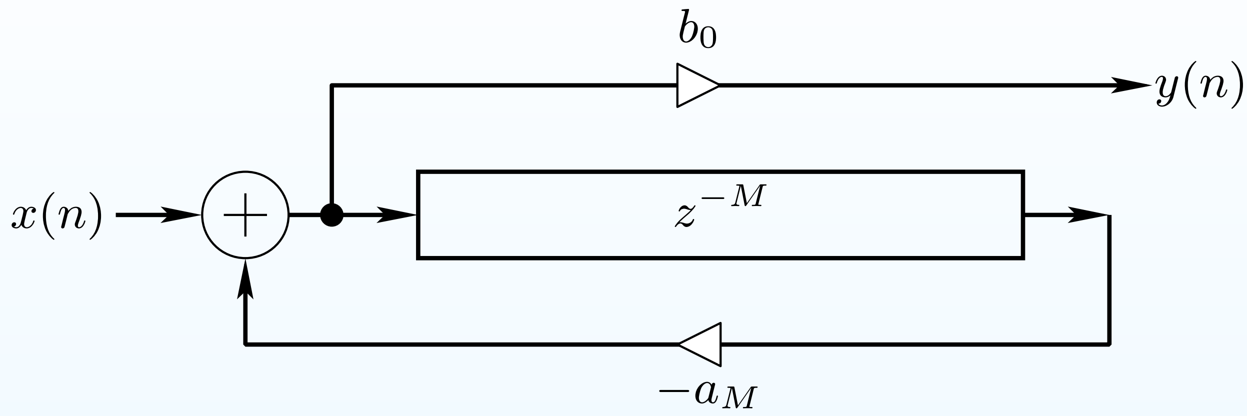
Two tape machines configured to produce a *flanging effect*.

Flanging Model



The basic flanger effect.

Feedback Comb Filter



Transfer Function

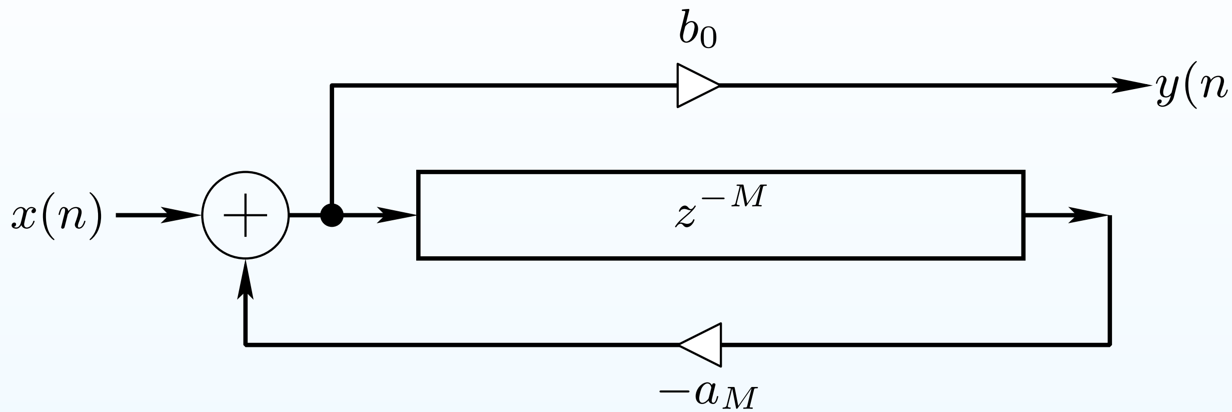
$$H(z) = \frac{b_0}{1 + a_M z^{-M}}$$

Frequency Response

$$H(e^{j\omega T}) = \frac{b_0}{1 + a_M e^{-jM\omega T}}$$

- Models plane waves *between two walls* (Moorer ’79)
- Models waves on a *terminated* ideal string

Feedback Comb Filter



Transfer Function

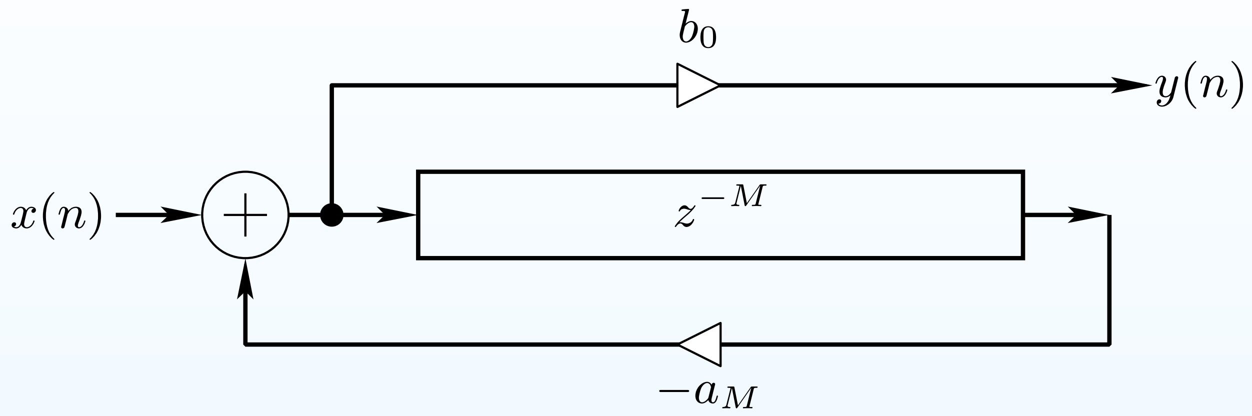
$$H(z) = \frac{b_0}{1 + a_M z^{-M}}$$

Frequency Response

$$H(e^{j\omega T}) = \frac{b_0}{1 + a_M e^{-jM\omega T}}$$

- Models plane waves *between two walls* (Moorer ’79)
- Models waves on a *terminated* ideal string

Feedback Comb Filter



Transfer Function

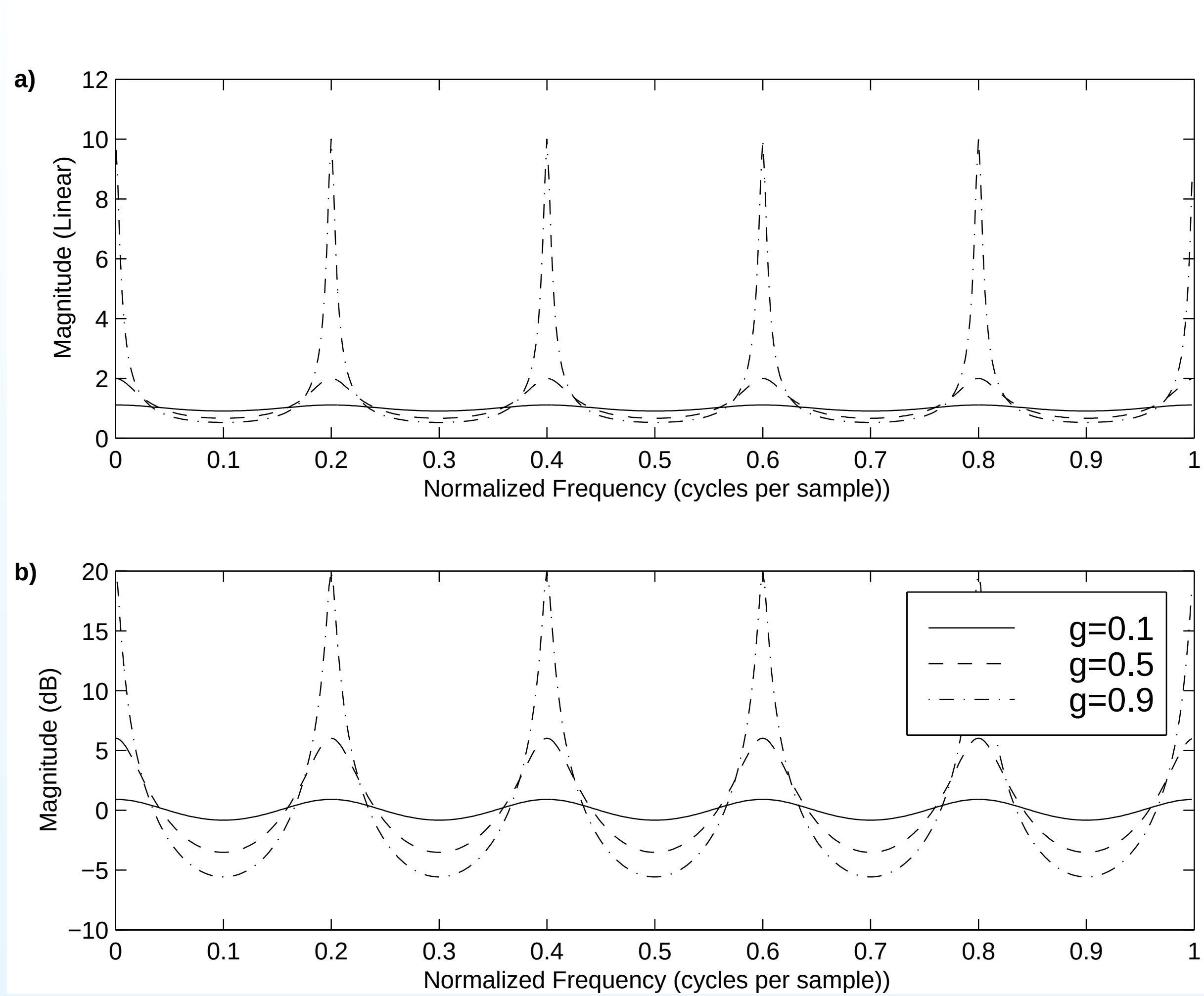
$$H(z) = \frac{b_0}{1 + a_M z^{-M}}$$

Frequency Response

$$H(e^{j\omega T}) = \frac{b_0}{1 + a_M e^{-jM\omega T}}$$

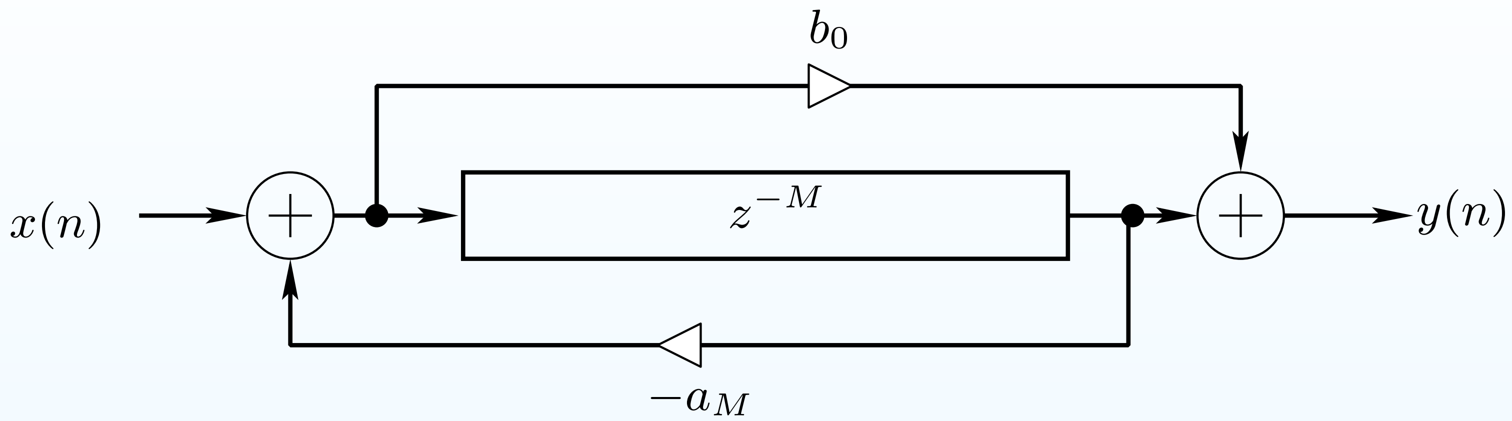
- Models plane waves *between two walls* (Moorer '79)
- Models waves on a *terminated* ideal string

Feedback Comb-Filter Amplitude Response



- $H(z) = \frac{1}{1-gz^{-M}}, \quad M = 5, \quad g = 0.1, 0.5, 0.9$
- $G(\omega) \stackrel{\Delta}{=} \left|H(e^{j\omega T})\right| = \left|\frac{1}{1-ge^{-jM\omega T}}\right|$

Schroeder Allpass Filters

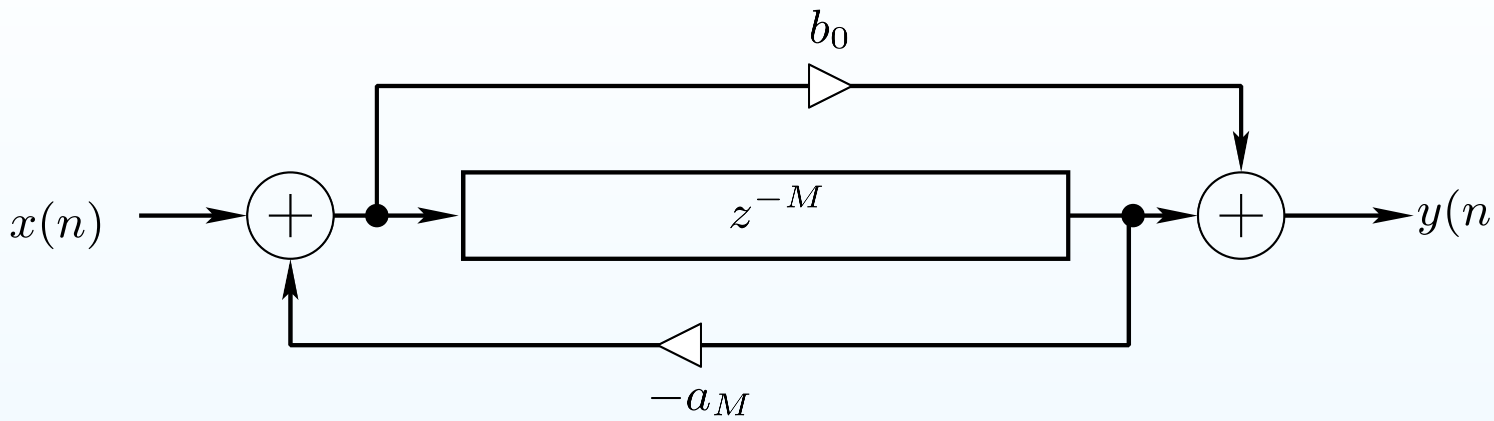


- Used extensively in artificial reverberation (since 1961)
- Transfer function:

$$H(z) = \frac{b_0 + z^{-M}}{1 + a_M z^{-M}}$$

- For allpass, set $b_0 = \overline{a_M}$

Schroeder Allpass Filters

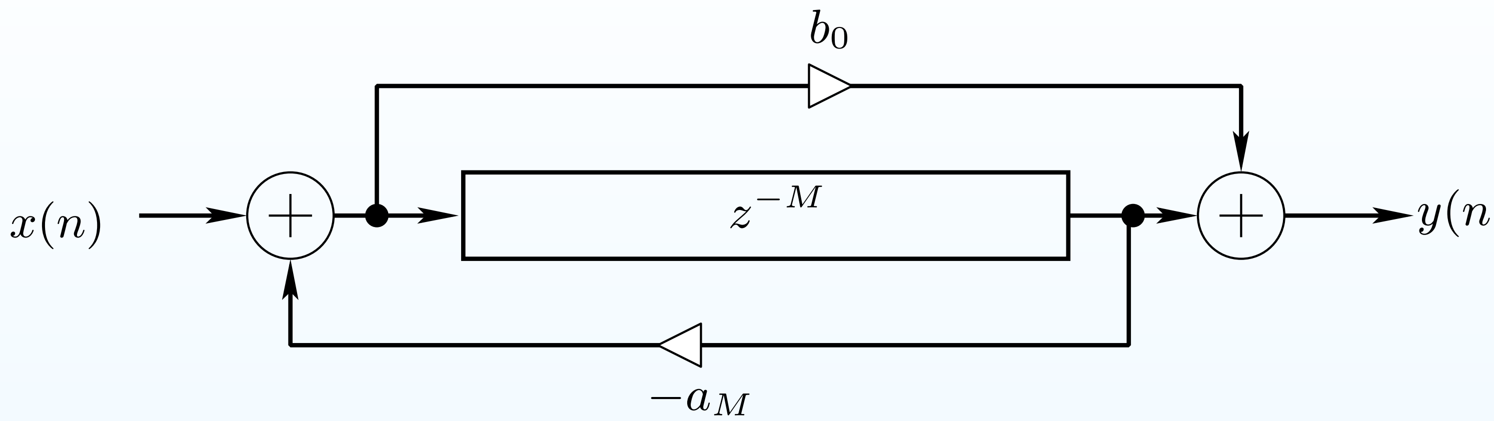


- Used extensively in artificial reverberation (since 1961)
- Transfer function:

$$H(z) = \frac{b_0 + z^{-M}}{1 + a_M z^{-M}}$$

- For allpass, set $b_0 = \overline{a_M}$

Schroeder Allpass Filters

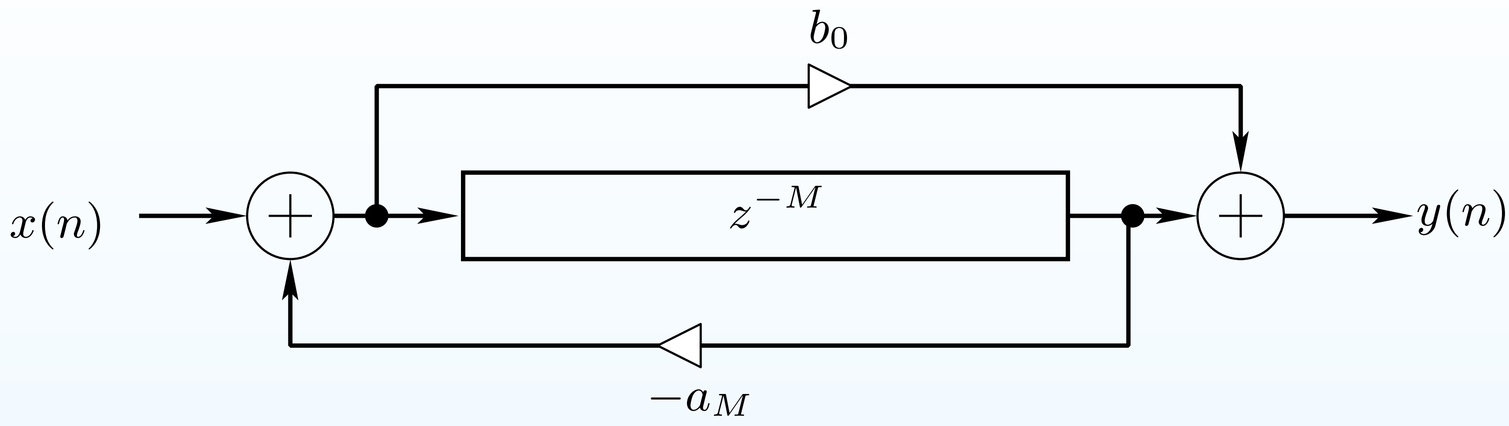


- Used extensively in artificial reverberation (since 1961)
- Transfer function:

$$H(z) = \frac{b_0 + z^{-M}}{1 + a_M z^{-M}}$$

- For allpass, set $b_0 = \overline{a_M}$

Schroeder Allpass Filters



- Used extensively in artificial reverberation (since 1961)
- Transfer function:

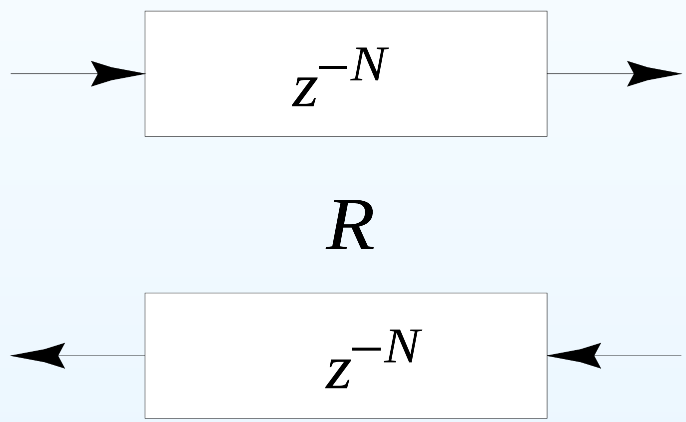
$$H(z) = \frac{b_0 + z^{-M}}{1 + a_M z^{-M}}$$

- For allpass, set $b_0 = \overline{a_M}$

Digital Waveguide Models

Digital Waveguide Models (1985)

Lossless digital waveguide $\stackrel{\Delta}{=}$ *bidirectional delay line*
at some *wave impedance* R



Useful for efficient models of

- strings
- bores
- plane waves
- conical waves

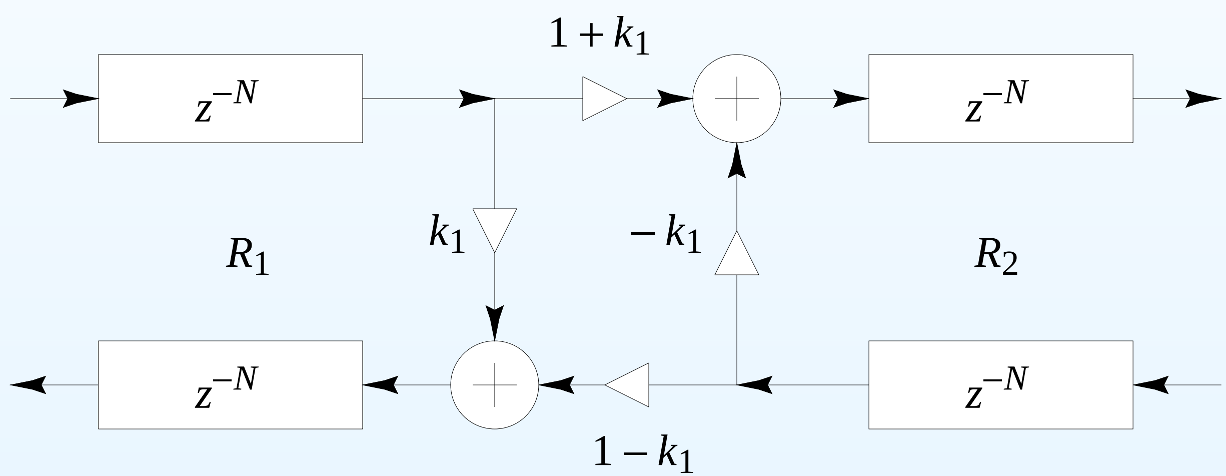
Early DAFx
Delay Effects
Waveguide Models
- Digital Waveguide
- Signal Scattering
- Plucked String
- Struck String
- Karplus Strong
- EKS Algorithm
- Physical Excitation
- Pick Position FFCF
- PLPC Cello
- Clarinet
- Wind Examples
- Bowed Strings
- Distortion Guitar

Commuted Synthesis
Summary
Related Topics

Signal Scattering

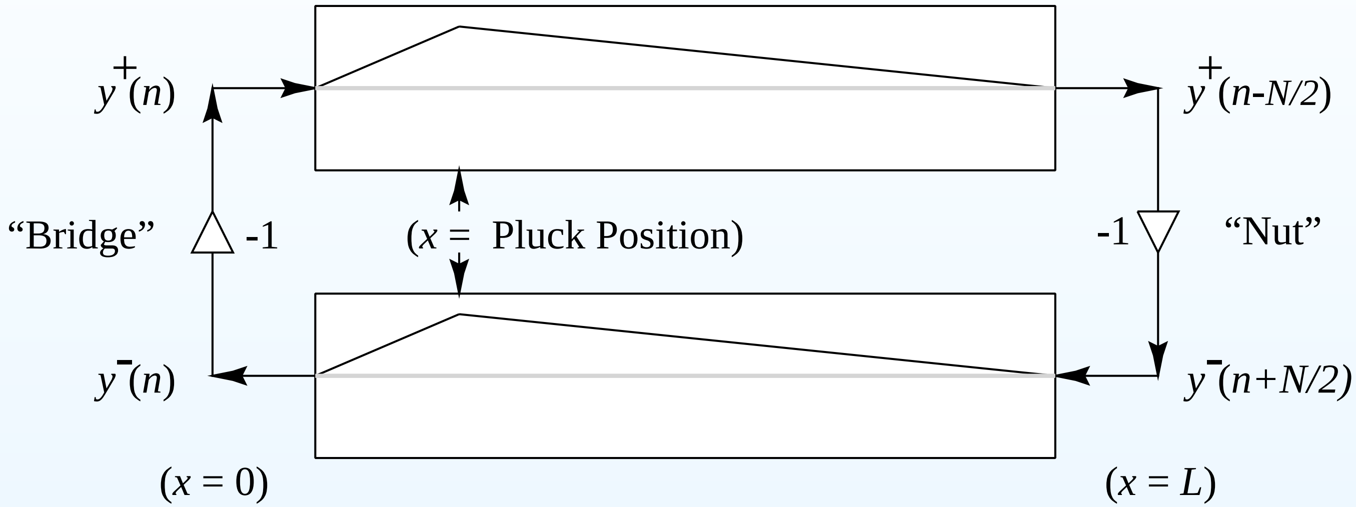
Signal *scattering* is caused by a *change* in wave impedance R:

$$k_1 = \frac{R_2 - R_1}{R_2 + R_1}$$



If the wave impedance changes *every spatial sample*, the Kelly-Lochbaum vocal-tract model results.

Ideal Plucked String (Displacement Waves)



- Load each delay line with *half* of initial string displacement
- Sum of upper and lower delay lines = string displacement

Early DAFx

Delay Effects

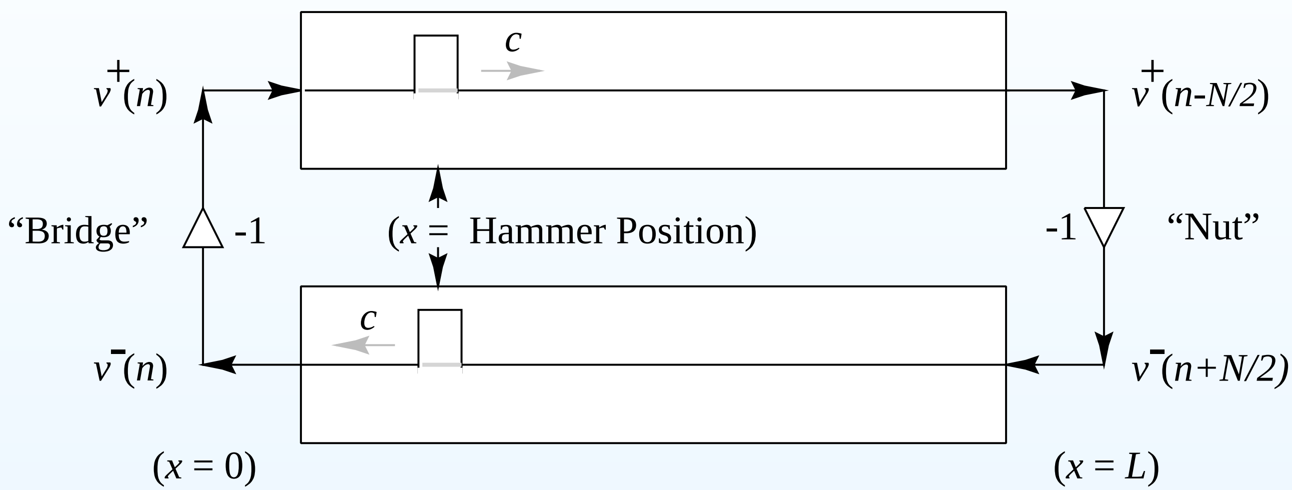
Waveguide Models
- Digital Waveguide
- Signal Scattering
- Plucked String
- Struck String
- Karplus Strong
- EKS Algorithm
- Physical Excitation
- Pick Position FFCF
- PLPC Cello
- Clarinet
- Wind Examples
- Bowed Strings
- Distortion Guitar

Commuted Synthesis

Summary

Related Topics

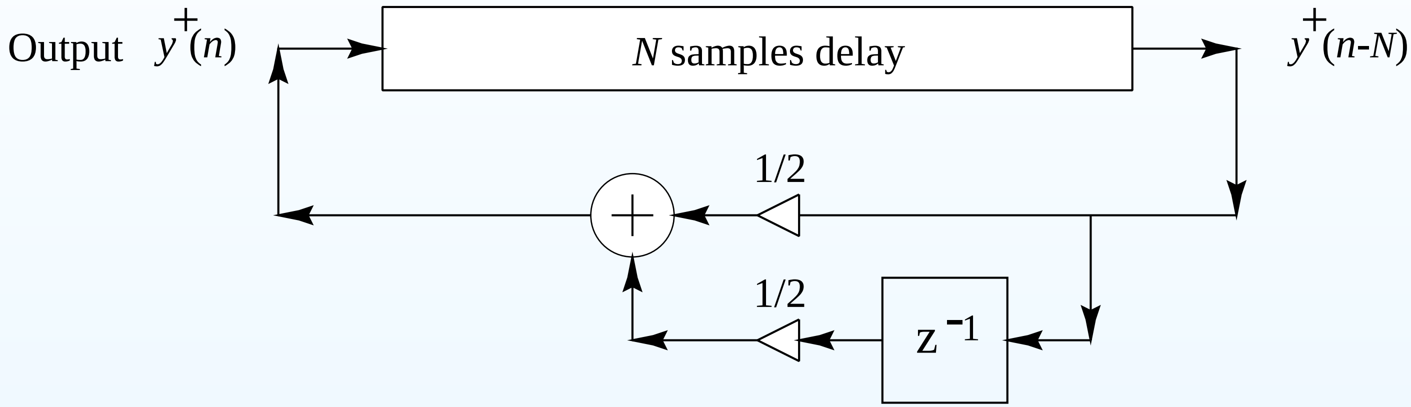
Ideal Struck String (Velocity Waves)



Hammer strike = *momentum transfer* = velocity step:

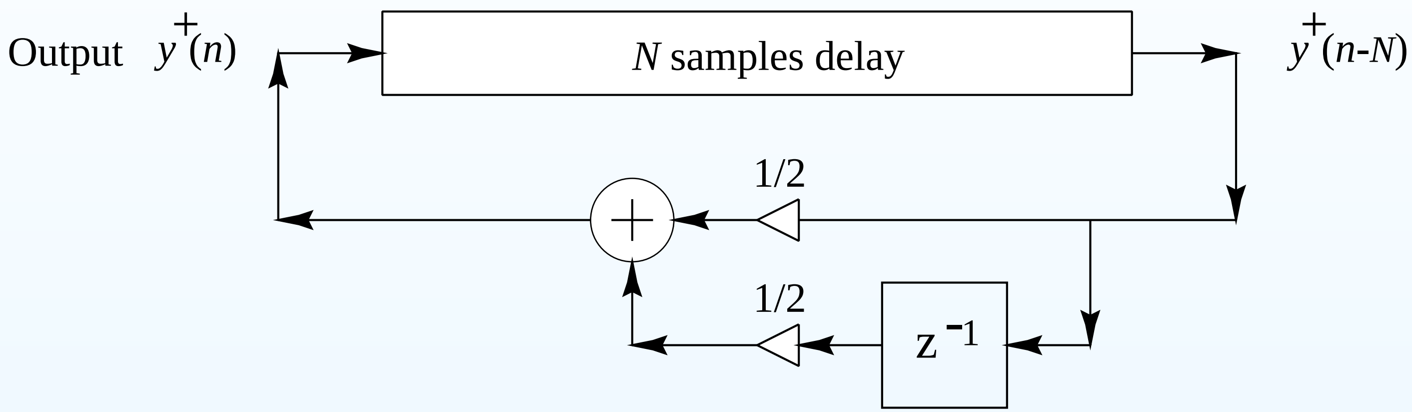
$$m_h v_h(0-) = (m_h + m_s) v_s(0+)$$

Karplus-Strong (KS) Algorithm (1983)



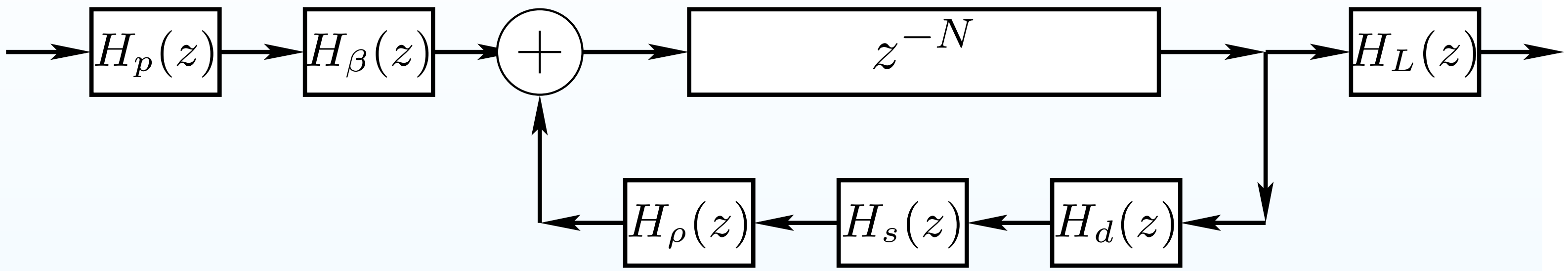
- Discovered (1978) as “self-modifying wavetable synthesis”
- Wavetable is preferably initialized with random numbers

Karplus-Strong (KS) Algorithm (1983)



- Discovered (1978) as “self-modifying wavetable synthesis”
- Wavetable is preferably initialized with random numbers

EKS Algorithm (Jaffe-Smith 1983)



$$N = \text{pitch period } (2\times \text{ string length}) \text{ in samples}$$

$$H_p(z) = \frac{1-p}{1-p\,z^{-1}} = \text{pick-direction lowpass filter}$$

$$H_\beta(z) = 1 - z^{-\beta N} = \text{pick-position comb filter, } \beta \in (0,1)$$

$$H_d(z) = \text{string-damping filter (one/two poles/zeros typical)}$$

$$H_s(z) = \text{string-stiffness allpass filter (several poles and zeros)}$$

$$H_\rho(z) = \frac{\rho(N) - z^{-1}}{1-\rho(N)\,z^{-1}} = \text{first-order string-tuning allpass filter}$$

$$H_L(z) = \frac{1-R_L}{1-R_L\,z^{-1}} = \text{dynamic-level lowpass filter}$$

STK EKS Sound Examples

- Synthesis Tool Kit (STK) by Perry Cook, Gary Scavone, and others — distributed by CCRMA:
  `Google` search: *STK ToolKit*

STK Plucked String: (WAV) (MP3)

- Plucked String 1: (WAV) (MP3)
- Plucked String 2: (WAV) (MP3)
- Plucked String 3: (WAV) (MP3)

EKS Sound Example (1988)

Bach A-Minor Concerto—Orchestra Part: (WAV) (MP3)

- Executed in real time on one Motorola DSP56001 (20 MHz clock, 128K SRAM)

EKS Sound Example (1988)

Bach A-Minor Concerto—Orchestra Part: (WAV) (MP3)

- Executed in real time on one Motorola DSP56001 (20 MHz clock, 128K SRAM)

- Developed for the NeXT Computer introduction at Davies Symphony Hall, San Francisco, 1988

EKS Sound Example (1988)

Bach A-Minor Concerto—Orchestra Part: (WAV) (MP3)

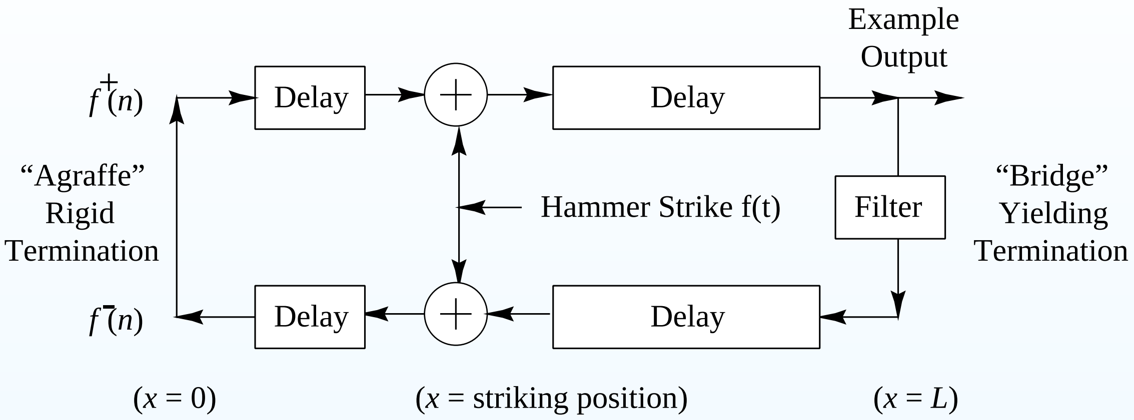
- Executed in real time on one Motorola DSP56001 (20 MHz clock, 128K SRAM)
- Developed for the NeXT Computer introduction at Davies Symphony Hall, San Francisco, 1988
- Solo violin part was played live by Dan Kobialka of the San Francisco Symphony

Example EKS Extension

Several of the Karplus-Strong algorithm extensions were based on its *physical interpretation*.

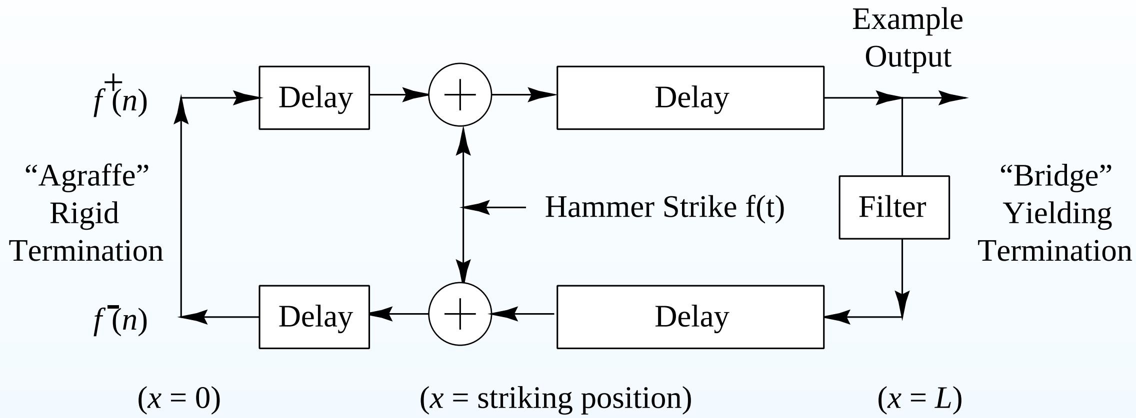
- Originally, transfer-function methods were used (1982)
- Below is a digital waveguide derivation

String Excited Externally at One Point



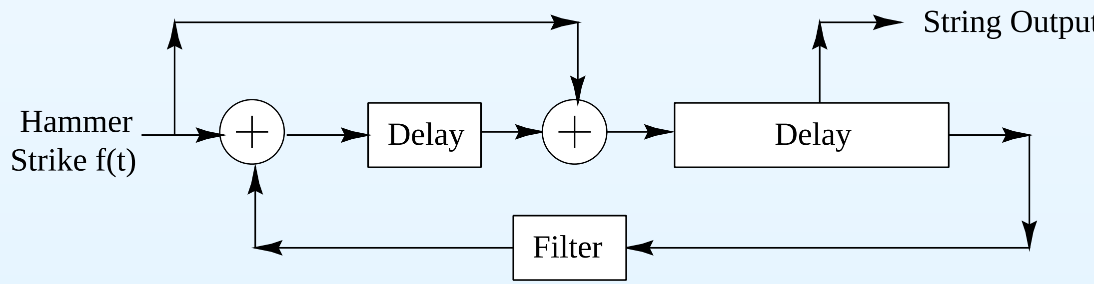
"Waveguide Canonical Form (1986)"

String Excited Externally at One Point

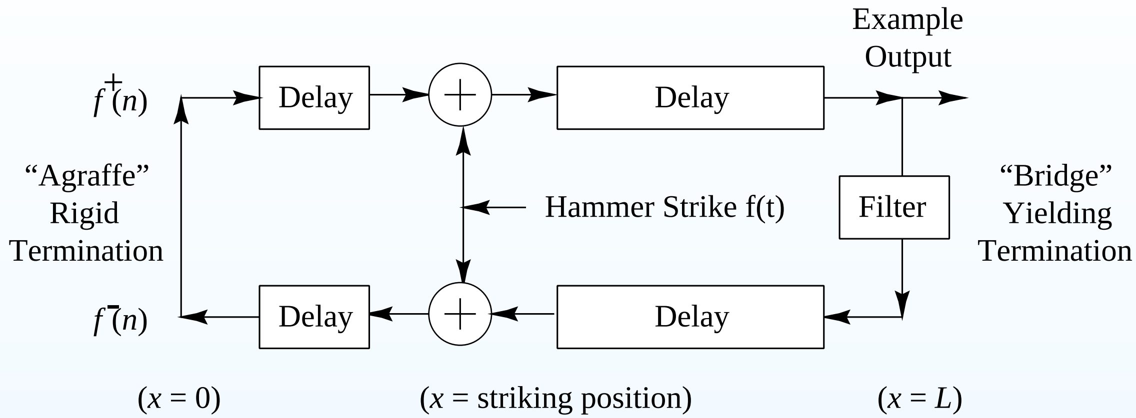


"Waveguide Canonical Form (1986)"

Equivalent System by Delay Consolidation:

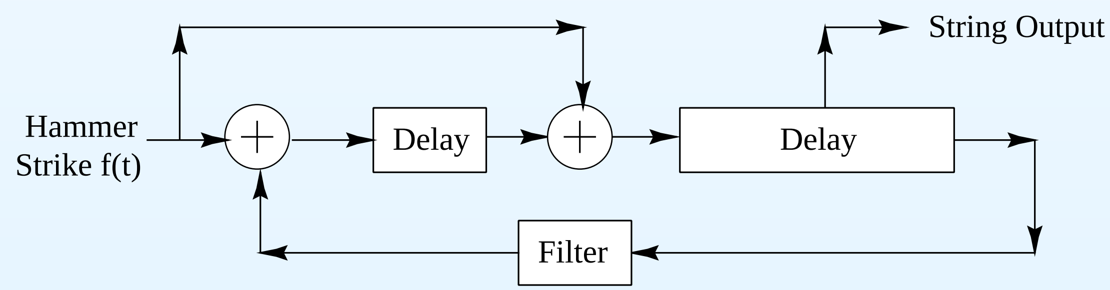


String Excited Externally at One Point



"Waveguide Canonical Form (1986)"

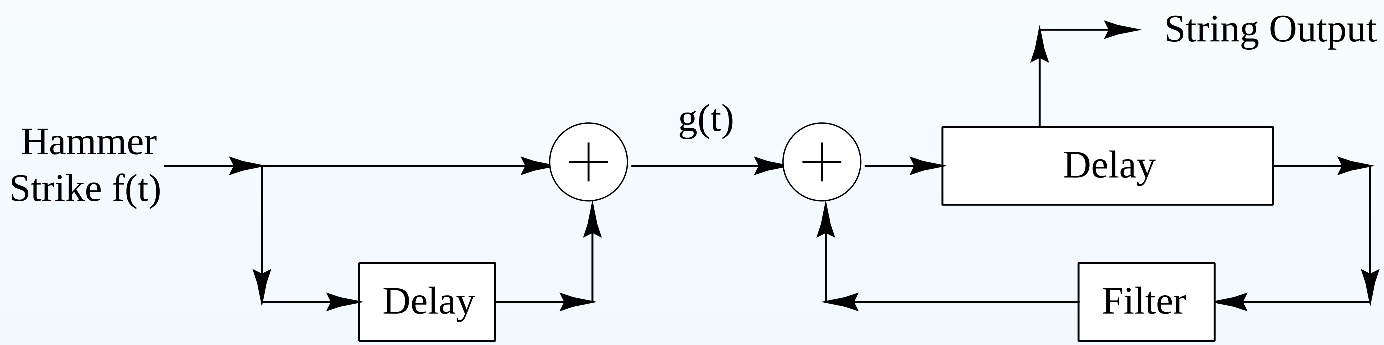
Equivalent System by Delay Consolidation:



Finally, we "pull out" the comb-filter component:

EKS “Pick Position” Extension

Equivalent System: Comb Filter Factored Out

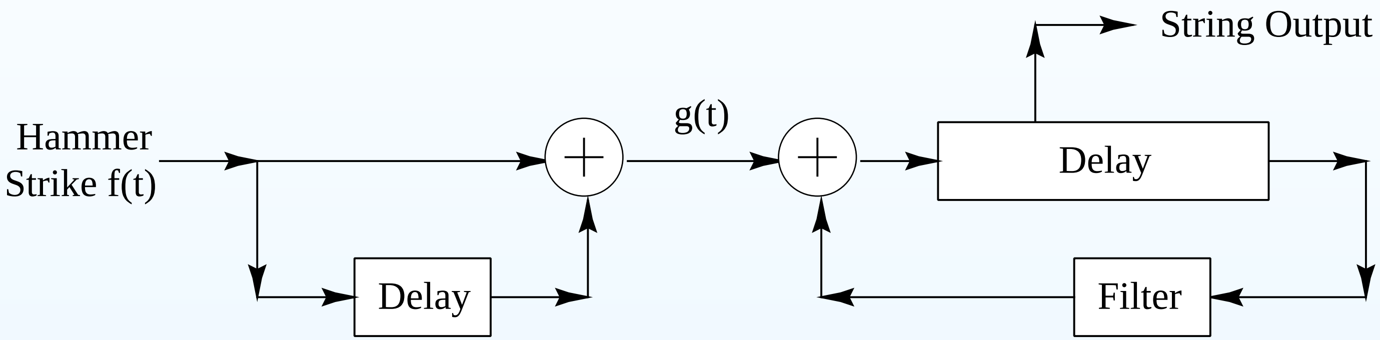


$$H(z) = z^{-N} \frac{1 + z^{-2M}}{1 - z^{-(2M+2N)}} = \left(1 + z^{-2M}\right) \frac{z^{-N}}{1 - z^{-(2M+2N)}}$$

- *Excitation Position* controlled by left delay-line length
- *Fundamental Frequency* controlled by right delay-line length
- “Transfer function modeling” based on a physical model (1982)

EKS "Pick Position" Extension

Equivalent System: Comb Filter Factored Out

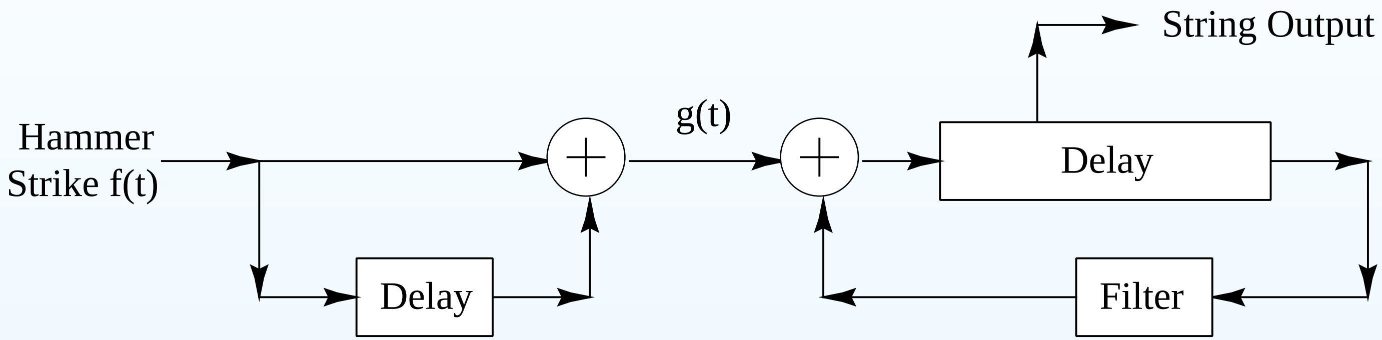


$$H(z) = z^{-N} \frac{1 + z^{-2M}}{1 - z^{-(2M+2N)}} = \left(1 + z^{-2M}\right) \frac{z^{-N}}{1 - z^{-(2M+2N)}}$$

- *Excitation Position* controlled by left delay-line length
- *Fundamental Frequency* controlled by right delay-line length
- "Transfer function modeling" based on a physical model (1982)

EKS "Pick Position" Extension

Equivalent System: Comb Filter Factored Out

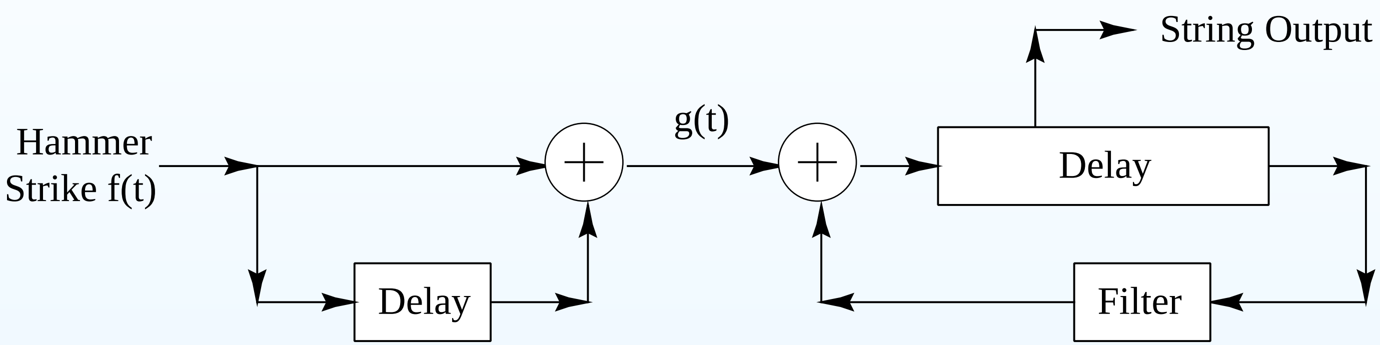


$$H(z) = z^{-N}\frac{1+z^{-2M}}{1-z^{-(2M+2N)}} = \left(1+z^{-2M}\right)\frac{z^{-N}}{1-z^{-(2M+2N)}}$$

- *Excitation Position* controlled by left delay-line length
- *Fundamental Frequency* controlled by right delay-line length
- "Transfer function modeling" based on a physical model (1982)

EKS "Pick Position" Extension

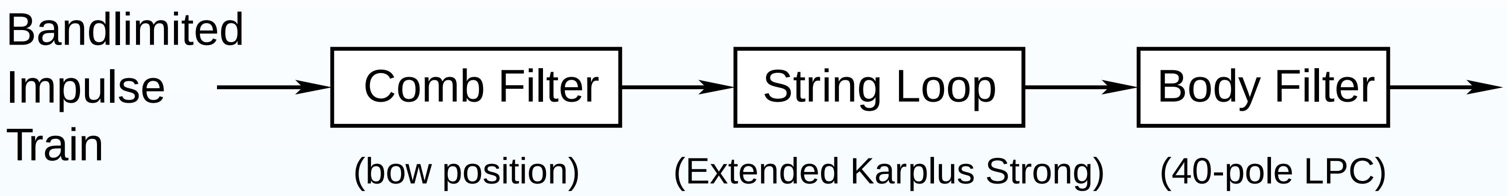
Equivalent System: Comb Filter Factored Out



$$H(z) = z^{-N} \frac{1 + z^{-2M}}{1 - z^{-(2M+2N)}} = \left(1 + z^{-2M}\right) \frac{z^{-N}}{1 - z^{-(2M+2N)}}$$

- *Excitation Position* controlled by left delay-line length
- *Fundamental Frequency* controlled by right delay-line length
- "Transfer function modeling" based on a physical model (1982)

PLPC Cello (1982)

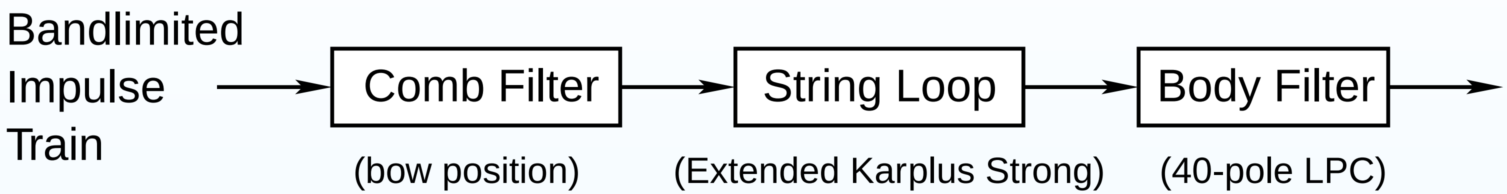


- Periodic LPC used to estimate string-loop filter
- Normal LPC used for body model (40 poles)
- Excitation = Bandlimited impulse train (Moorer 1975):

$$\sum_{k=1}^{K} \cos(k\omega_0 t) = \frac{\sin[(K+1/2)\omega_0 t]}{2\sin(\omega_0 t/2)} - \frac{1}{2}$$

- Bow-position simulation = variable-delay differencing comb filter (direct from physical interpretation)
- **Sound Example:**
  Moving Bow-Stroke Example: (WAV) (MP3)
  (Bowing point moves toward the “bridge”)

PLPC Cello (1982)



- Periodic LPC used to estimate string-loop filter
- Normal LPC used for body model (40 poles)
- Excitation = Bandlimited impulse train (Moorer 1975):

$$\sum_{k=1}^{K} \cos(k\omega_0 t) = \frac{\sin[(K+1/2)\omega_0 t]}{2\sin(\omega_0 t/2)} - \frac{1}{2}$$

- Bow-position simulation = variable-delay differencing comb filter (direct from physical interpretation)
- **Sound Example:**
  Moving Bow-Stroke Example: (WAV) (MP3)
  (Bowing point moves toward the "bridge")

Early DAFx

Delay Effects

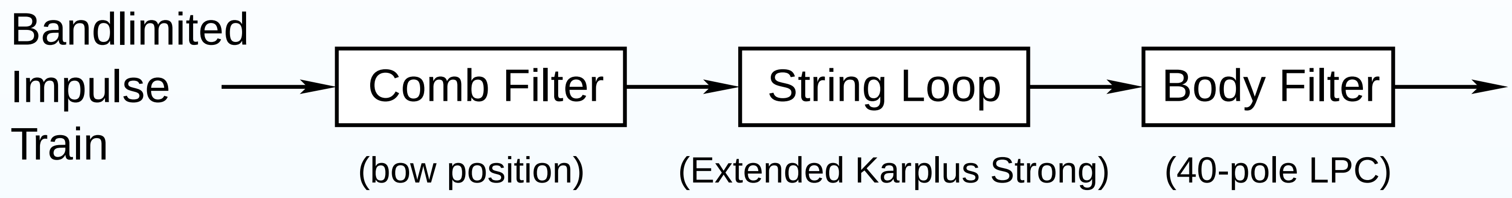
Waveguide Models
- Digital Waveguide
- Signal Scattering
- Plucked String
- Struck String
- Karplus Strong
- EKS Algorithm
- Physical Excitation
- Pick Position FFCF
- PLPC Cello
- Clarinet
- Wind Examples
- Bowed Strings
- Distortion Guitar

Commuted Synthesis

Summary

Related Topics

PLPC Cello (1982)

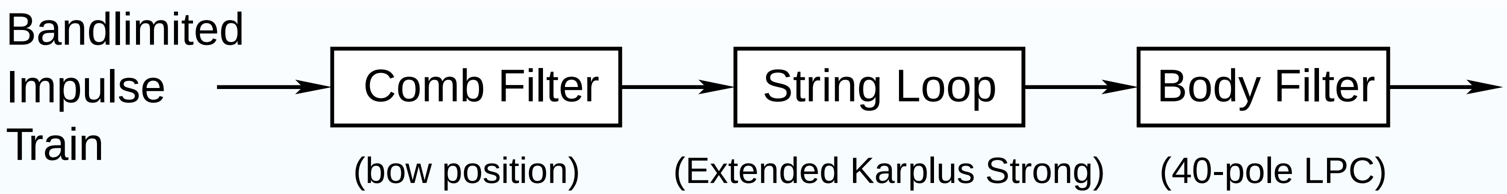


- Periodic LPC used to estimate string-loop filter
- Normal LPC used for body model (40 poles)
- Excitation = Bandlimited impulse train (Moorer 1975):

$$\sum_{k=1}^{K} \cos(k\omega_0 t) = \frac{\sin[(K+1/2)\omega_0 t]}{2\sin(\omega_0 t/2)} - \frac{1}{2}$$

- Bow-position simulation = variable-delay differencing comb filter (direct from physical interpretation)
- **Sound Example:**
  Moving Bow-Stroke Example: (WAV) (MP3)
  (Bowing point moves toward the "bridge")

PLPC Cello (1982)

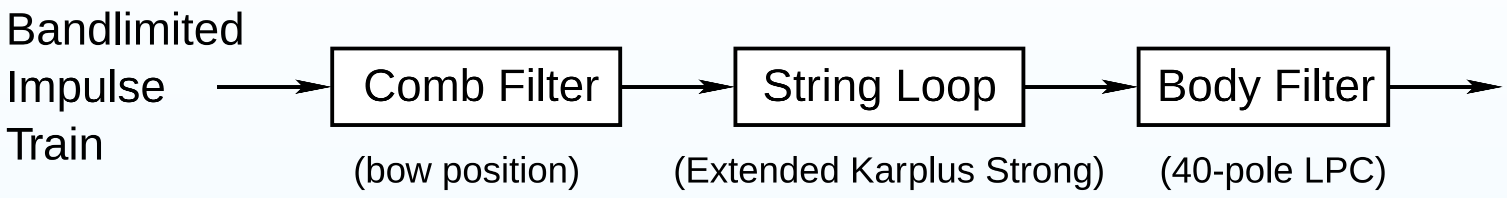


- Periodic LPC used to estimate string-loop filter
- Normal LPC used for body model (40 poles)
- Excitation = Bandlimited impulse train (Moorer 1975):

$$\sum_{k=1}^{K} \cos(k\omega_0 t) = \frac{\sin[(K+1/2)\omega_0 t]}{2\sin(\omega_0 t/2)} - \frac{1}{2}$$

- Bow-position simulation = variable-delay differencing comb filter (direct from physical interpretation)
- **Sound Example:**
  Moving Bow-Stroke Example: (WAV) (MP3)
  (Bowing point moves toward the “bridge”)

PLPC Cello (1982)

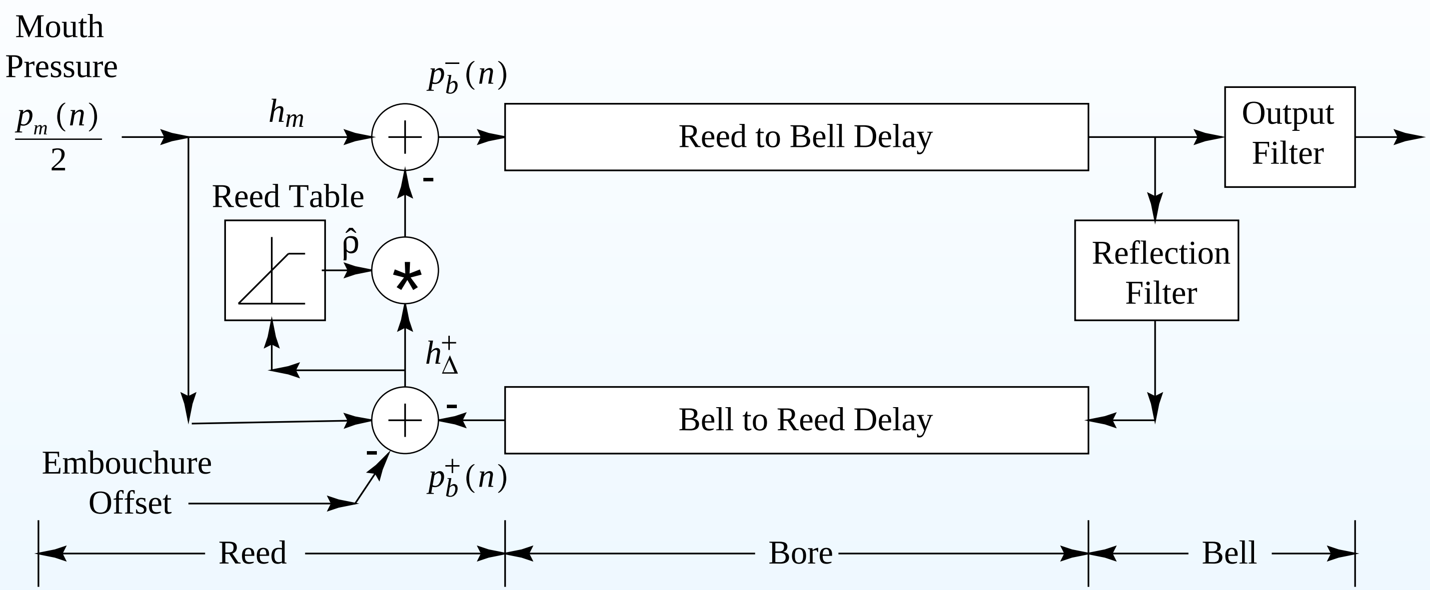


- Periodic LPC used to estimate string-loop filter
- Normal LPC used for body model (40 poles)
- Excitation = Bandlimited impulse train (Moorer 1975):

$$\sum_{k=1}^{K} \cos(k\omega_0 t) = \frac{\sin[(K+1/2)\omega_0 t]}{2\sin(\omega_0 t/2)} - \frac{1}{2}$$

- Bow-position simulation = variable-delay differencing comb filter (direct from physical interpretation)
- **Sound Example:**
  Moving Bow-Stroke Example: (WAV) (MP3)
  (Bowing point moves toward the “bridge”)

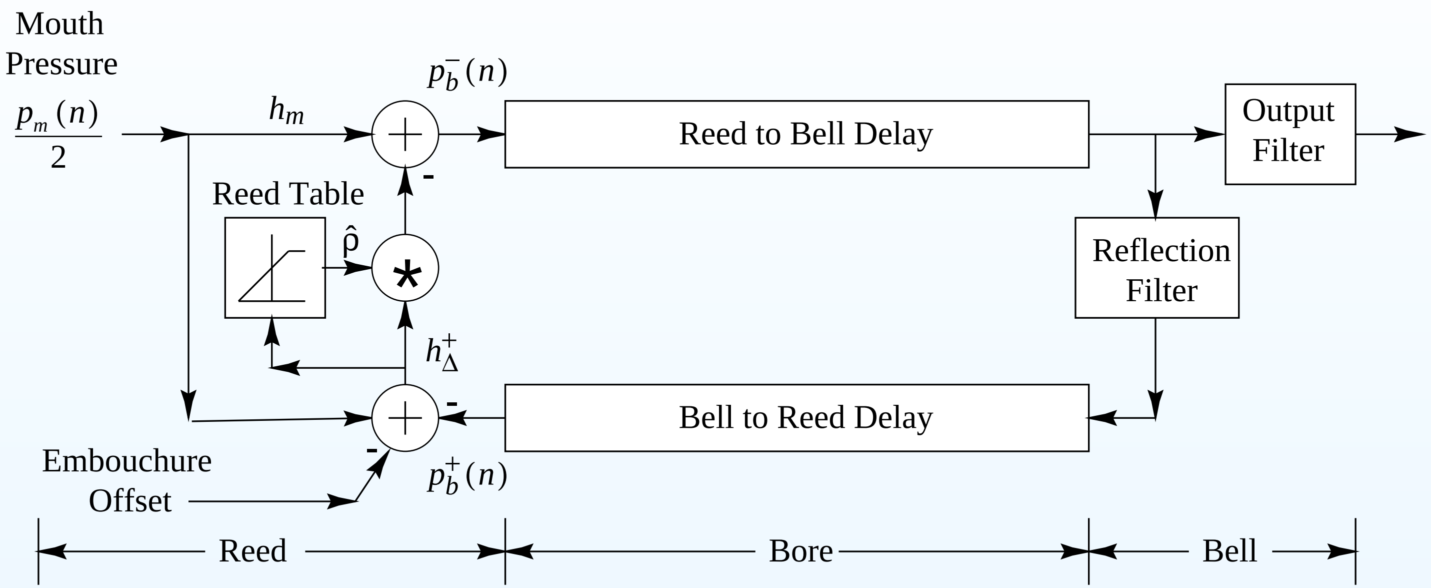
Digital Waveguide Single Reed, Cylindrical Bore Model (1986)



Digital waveguide clarinet

- Control variable = mouth half-pressure
- Total reed cost = two subtractions, one multiply, and one table lookup per sample

Digital Waveguide Single Reed, Cylindrical Bore Model (1986)



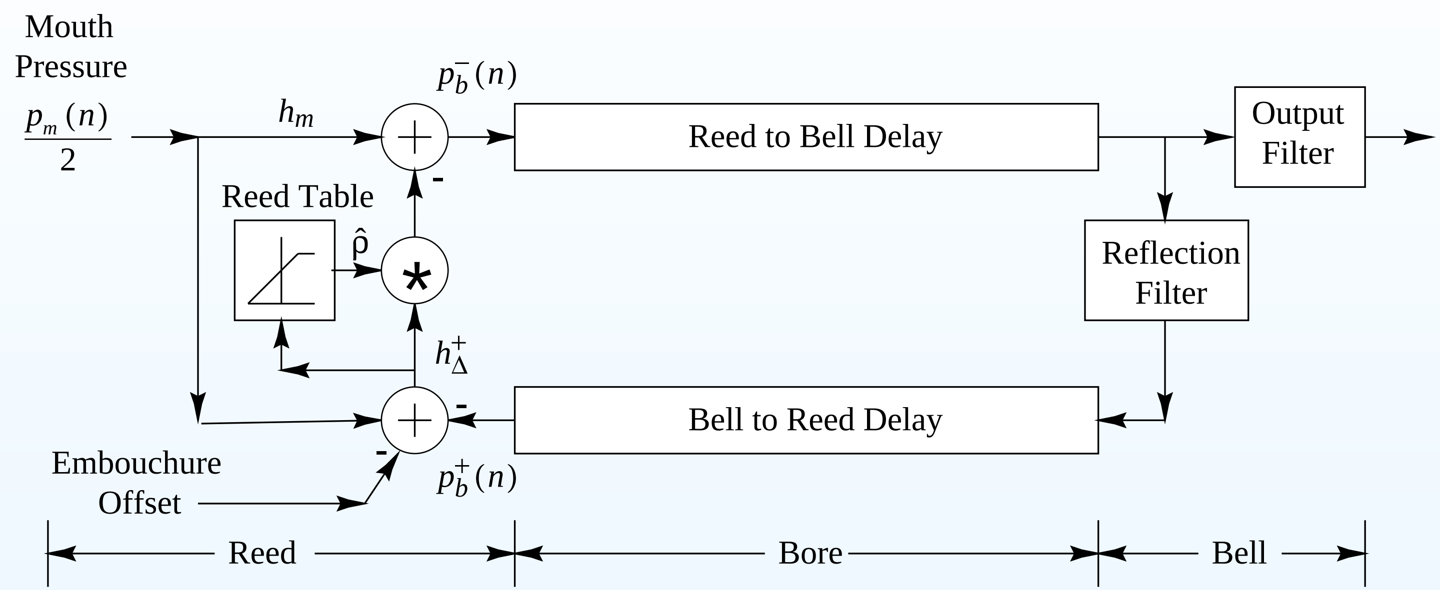
Digital waveguide clarinet

- Control variable = mouth half-pressure
- Total reed cost = two subtractions, one multiply, and one table lookup per sample

Early DAFx
Delay Effects
Waveguide Models
- Digital Waveguide
- Signal Scattering
- Plucked String
- Struck String
- Karplus Strong
- EKS Algorithm
- Physical Excitation
- Pick Position FFCF
- PLPC Cello
- Clarinet
- Wind Examples
- Bowed Strings
- Distortion Guitar

Commuted Synthesis
Summary
Related Topics

Digital Waveguide Single Reed, Cylindrical Bore Model (1986)



Digital waveguide clarinet

- Control variable = mouth half-pressure
- Total reed cost = two subtractions, one multiply, and one table lookup per sample

Digital Waveguide Wind Instrument Sound Examples

- STK Clarinet: (WAV) (MP3)
  `Google` search: *STK clarinet*
  - Synthesis Tool Kit (STK) by Perry Cook, Gary Scavone, and others — distributed by CCRMA:
    `Google` search: *STK ToolKit*
- Staccato Systems Slide Flute
  (based on STK flute, ca. 1995): (WAV) (MP3)
- Yamaha VL1 "Virtual Lead" synthesizer demos (1994):
  - Shakuhachi: (WAV) (MP3)
  - Oboe and Bassoon: (WAV) (MP3)
  - Tenor Saxophone: (WAV) (MP3)

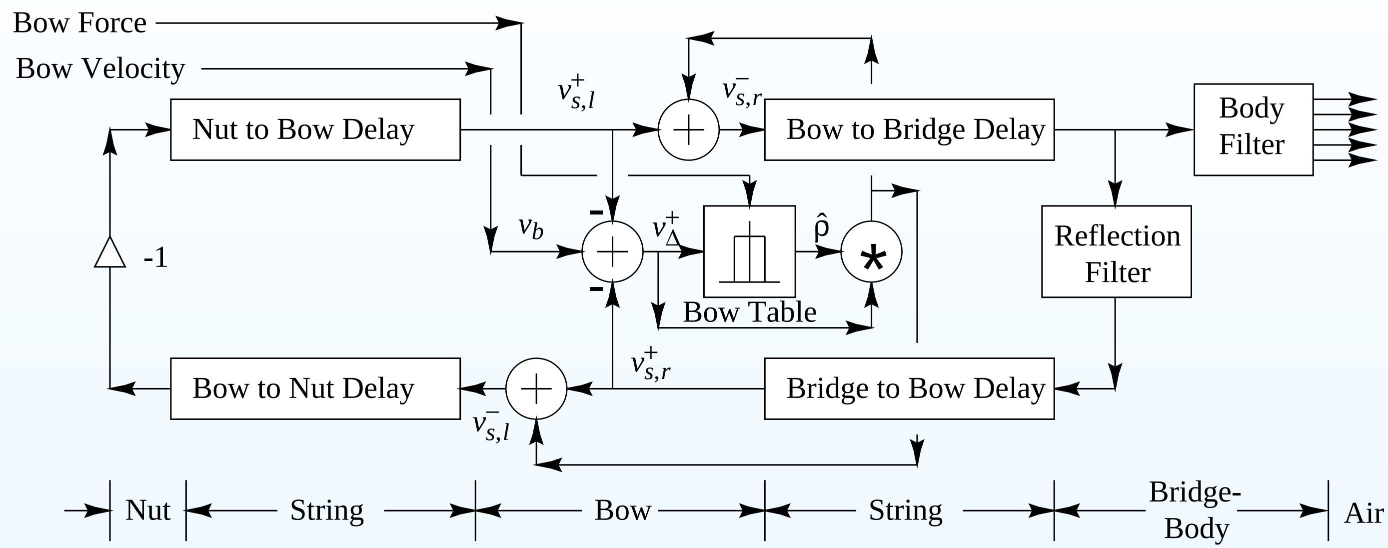
Digital Waveguide Wind Instrument Sound Examples

- STK Clarinet: (WAV) (MP3)
  `Google` search: *STK clarinet*
  - Synthesis Tool Kit (STK) by Perry Cook, Gary Scavone, and others — distributed by CCRMA:
    `Google` search: *STK ToolKit*
- Staccato Systems Slide Flute
  (based on STK flute, ca. 1995): (WAV) (MP3)
- Yamaha VL1 "Virtual Lead" synthesizer demos (1994):
  - Shakuhachi: (WAV) (MP3)
  - Oboe and Bassoon: (WAV) (MP3)
  - Tenor Saxophone: (WAV) (MP3)

Digital Waveguide Wind Instrument Sound Examples

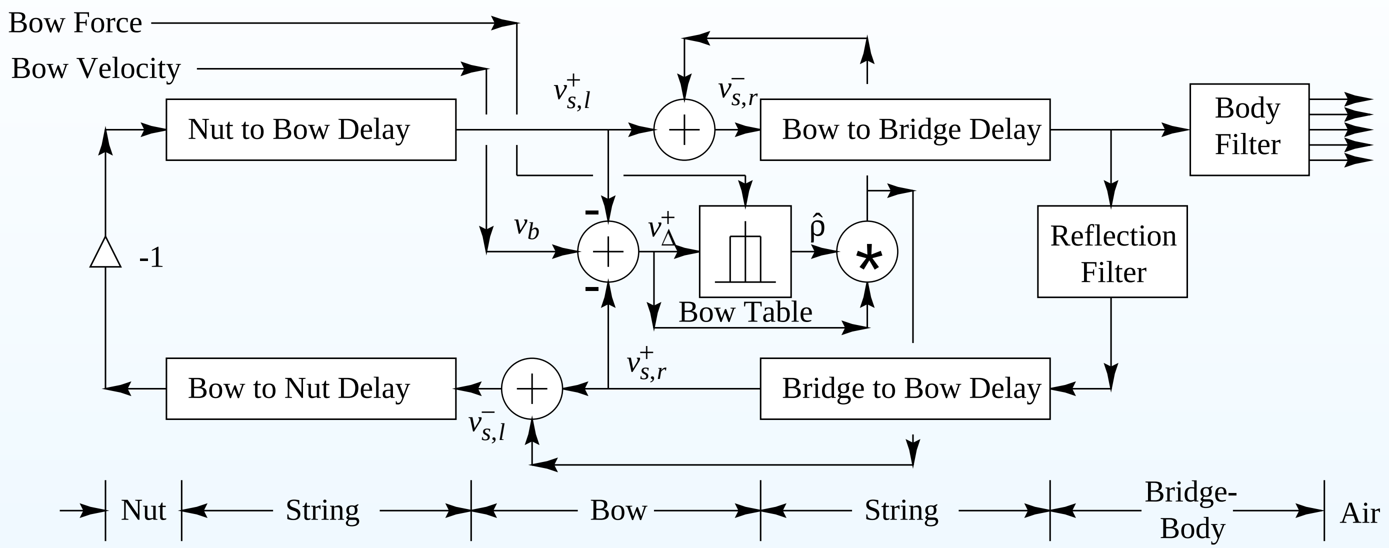
- STK Clarinet: (WAV) (MP3)
  `Google` search: *STK clarinet*
  - Synthesis Tool Kit (STK) by Perry Cook, Gary Scavone, and others — distributed by CCRMA:
    `Google` search: *STK ToolKit*
- Staccato Systems Slide Flute
  (based on STK flute, ca. 1995): (WAV) (MP3)
- Yamaha VL1 “Virtual Lead” synthesizer demos (1994):
  - Shakuhachi: (WAV) (MP3)
  - Oboe and Bassoon: (WAV) (MP3)
  - Tenor Saxophone: (WAV) (MP3)

Digital Waveguide Bowed Strings (1986)



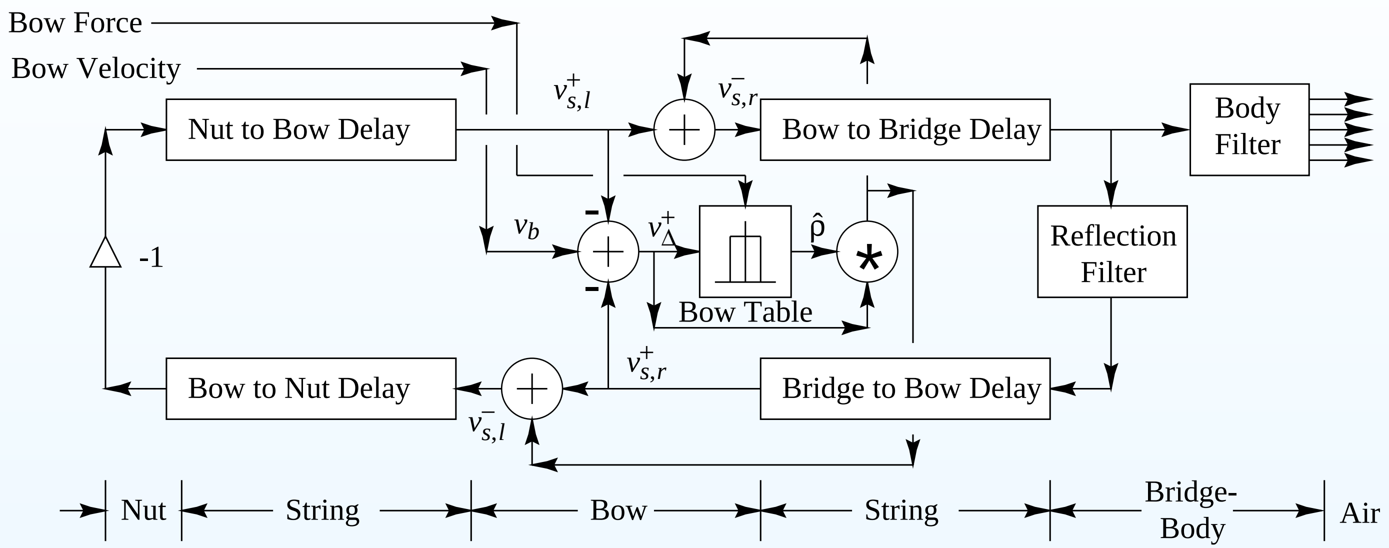
- Reflection filter summarizes all losses per period (due to bridge, bow, finger, etc.)
- Bow-string junction = *memoryless* lookup table (or segmented polynomial)

Digital Waveguide Bowed Strings (1986)



- Reflection filter summarizes all losses per period (due to bridge, bow, finger, etc.)
- Bow-string junction = *memoryless* lookup table (or segmented polynomial)

Digital Waveguide Bowed Strings (1986)



- Reflection filter summarizes all losses per period (due to bridge, bow, finger, etc.)
- Bow-string junction = *memoryless* lookup table (or segmented polynomial)

“Electric Cello” Sound Examples (Peder Larson)

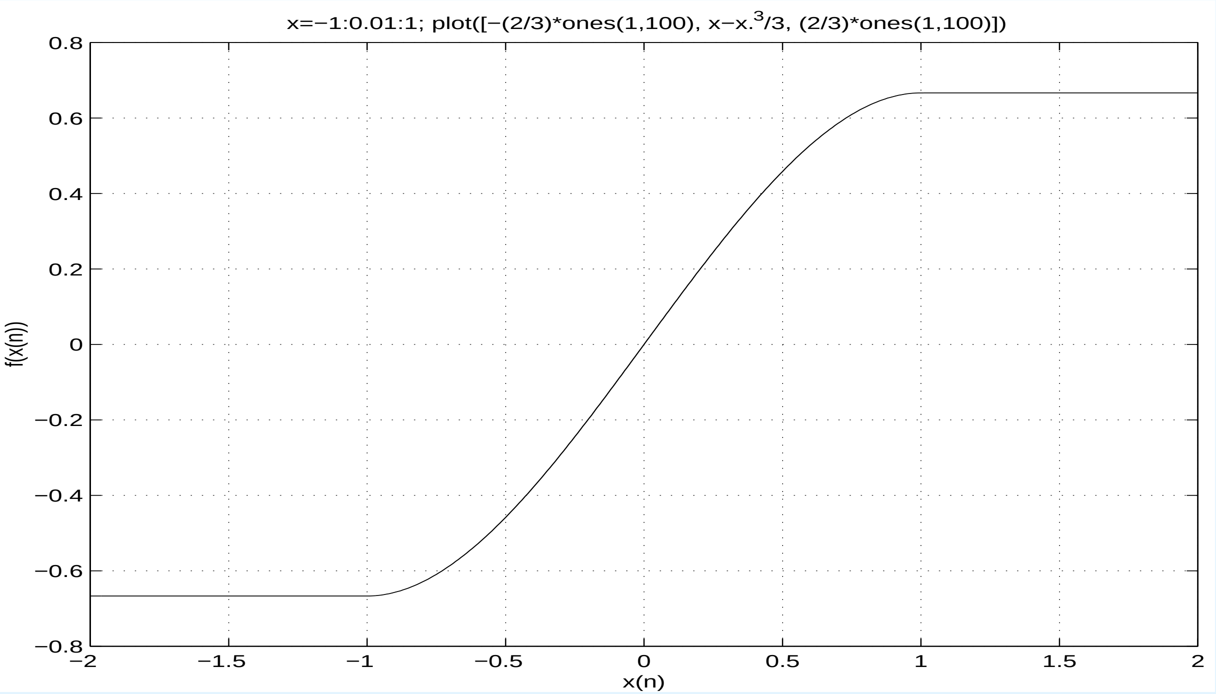
- Staccato Notes: (WAV) (MP3)
  (short strokes of high bow pressure, as from a bouncing bow)
- Bach’s First Suite for Unaccompanied Cello: (WAV) (MP3)

“Electric Cello” Sound Examples (Peder Larson)

- Staccato Notes: (WAV) (MP3)
  (short strokes of high bow pressure, as from a bouncing bow)
- Bach’s First Suite for Unaccompanied Cello: (WAV) (MP3)

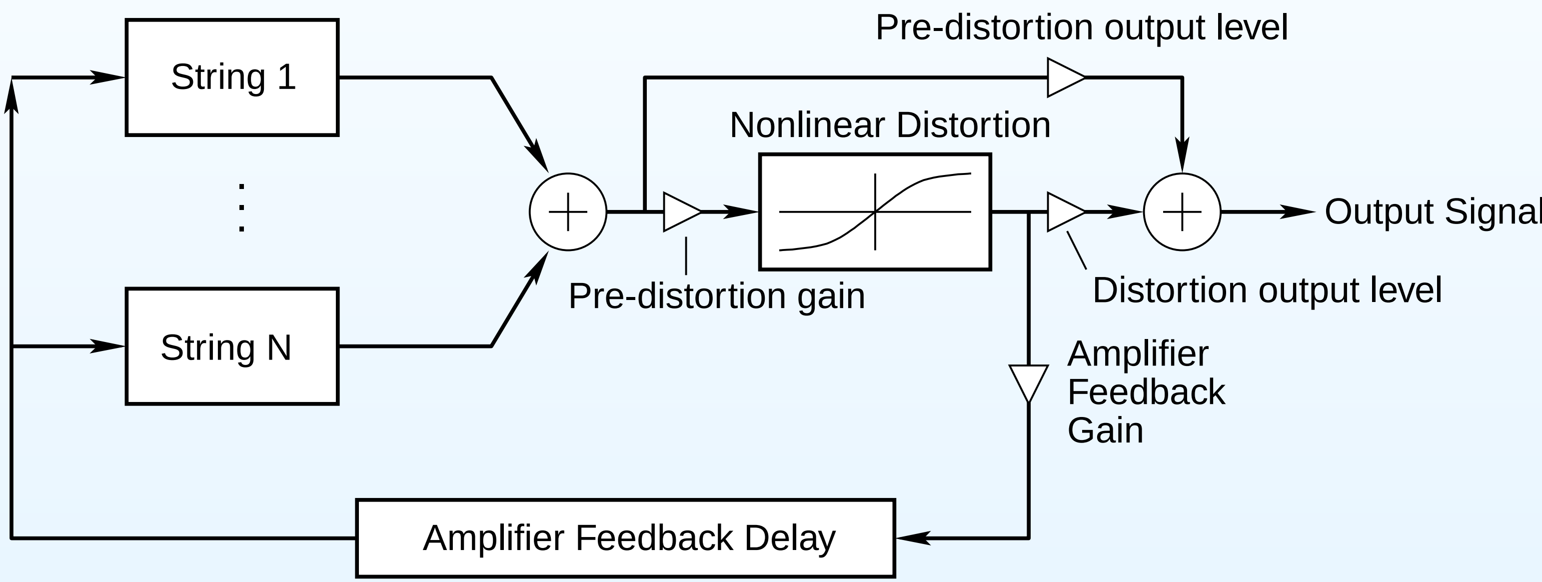
Soft Clipper

$$f(x) = \begin{cases} -\frac{2}{3}, & x \leq -1 \\ x - \frac{x^3}{3}, & -1 \leq x \leq 1 \\ \frac{2}{3}, & x \geq 1 \end{cases}$$



Amplifier Distortion + Amplifier Feedback

Sullivan 1990



Distortion output signal often further filtered by an *amplifier cabinet filter*, representing speaker cabinet, driver responses, etc.

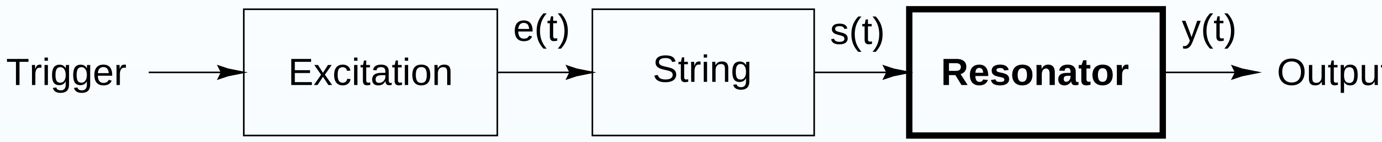
Distortion Guitar Sound Examples

(Stanford Sondius Project, ca. 1995)

- Distortion Guitar: (WAV) (MP3)
- Amplifier Feedback 1: (WAV) (MP3)
- Amplifier Feedback 2: (WAV) (MP3)

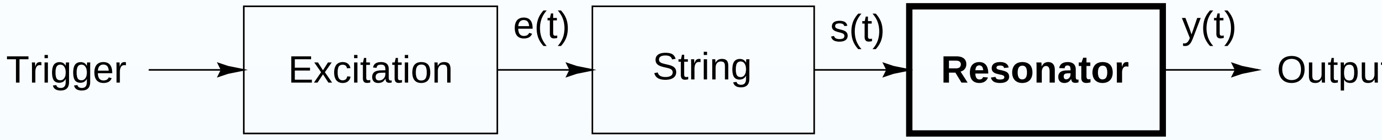
Commuted Waveguide Synthesis

Commuted Synthesis of Acoustic Strings (1993)



Schematic diagram of a stringed musical instrument.

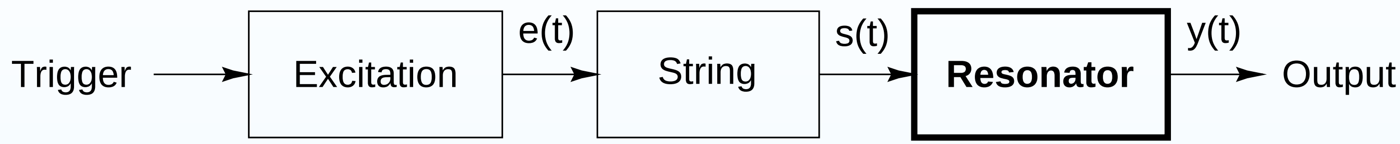
Commuted Synthesis of Acoustic Strings (1993)



Schematic diagram of a stringed musical instrument.

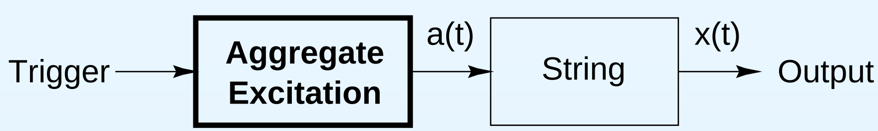
Equivalent diagram in the linear, time-invariant case.

Commuted Synthesis of Acoustic Strings (1993)



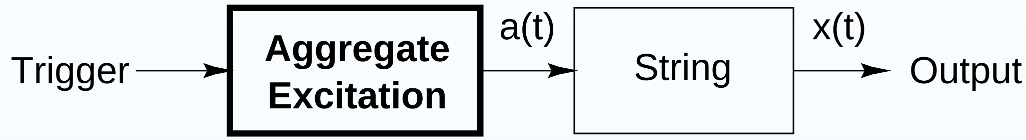
Schematic diagram of a stringed musical instrument.

Equivalent diagram in the linear, time-invariant case.

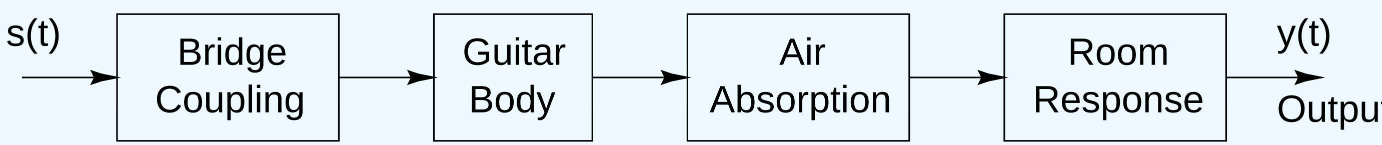


Use of an aggregate excitation given by the convolution of original excitation with the resonator impulse response.

Commuted Components



“Plucked Resonator” driving a String.



Possible components of a guitar resonator.

Sound Examples

Electric Guitar (Pick-Ups and/or Body-Model Added) (Stanford Sondius Project $\longrightarrow$ Staccato Systems, Inc. $\longrightarrow$ ADI, ca. 1995)

- Example 1: (WAV) (MP3)
- Example 2: (WAV) (MP3)
- Example 3: (WAV) (MP3)
- Virtual "wah-wah pedal": (WAV) (MP3)

Sound Examples

Electric Guitar (Pick-Ups and/or Body-Model Added) (Stanford Sondius Project $\longrightarrow$ Staccato Systems, Inc. $\longrightarrow$ ADI, ca. 1995)

- Example 1: (WAV) (MP3)
- Example 2: (WAV) (MP3)
- Example 3: (WAV) (MP3)
- Virtual “wah-wah pedal”: (WAV) (MP3)

STK Mandolin

- STK Mandolin 1: (WAV) (MP3)
- STK Mandolin 2: (WAV) (MP3)

Sound Examples

More Recent Acoustic Guitar

- Bach Prelude in E Major: (WAV) (MP3)
- Bach Loure in E Major: (WAV) (MP3)
- More examples
- Yet more examples

Virtual performance by Dr. Mikael Laurson, Sibelius Institute

Sound Examples

More Recent Acoustic Guitar

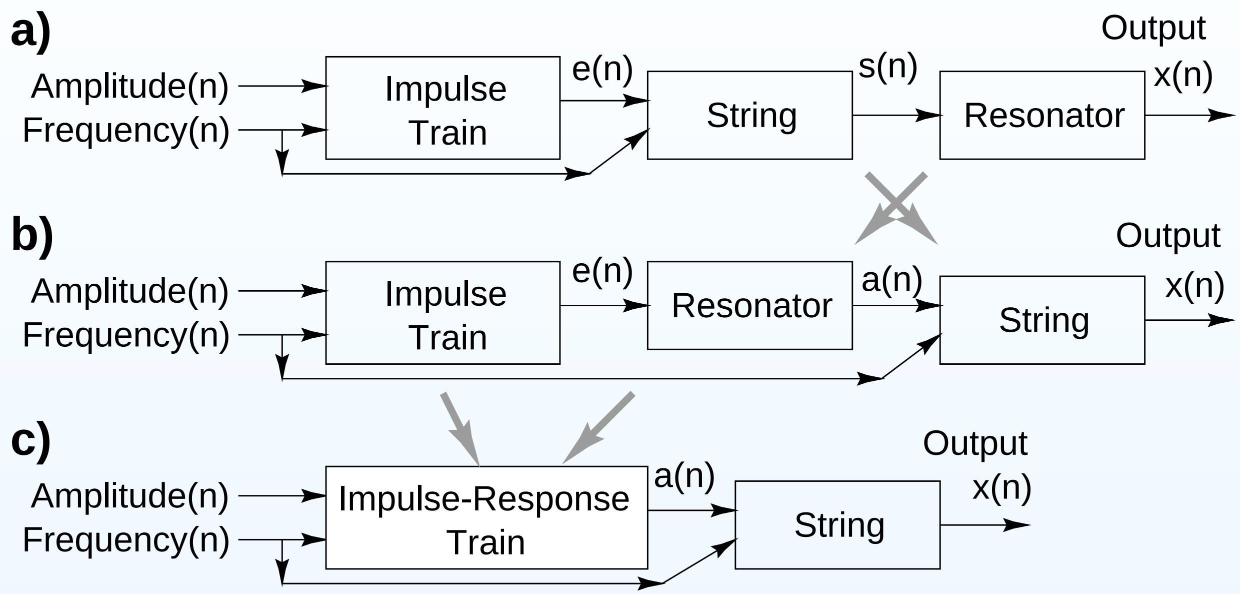
- Bach Prelude in E Major: (WAV) (MP3)
- Bach Loure in E Major: (WAV) (MP3)
- More examples
- Yet more examples

Virtual performance by Dr. Mikael Laurson, Sibelius Institute

Virtual guitar by Helsinki Univ. of Tech., Acoustics Lab[1]

[1] http://www.acoustics.hut.fi/

Commuted Synthesis of Linearized Violin



- Assumes *ideal Helmholtz motion* of string
- Sound Examples (Stanford Sondius project, ca. 1995):
  - Bass: (WAV) (MP3)
  - Cello: (WAV) (MP3)
  - Viola 1: (WAV) (MP3)
  - Viola 2: (WAV) (MP3)
  - Violin 1: (WAV) (MP3)
  - Violin 2: (WAV) (MP3)
  - Ensemble: (WAV) (MP3)

Early DAFx

Delay Effects

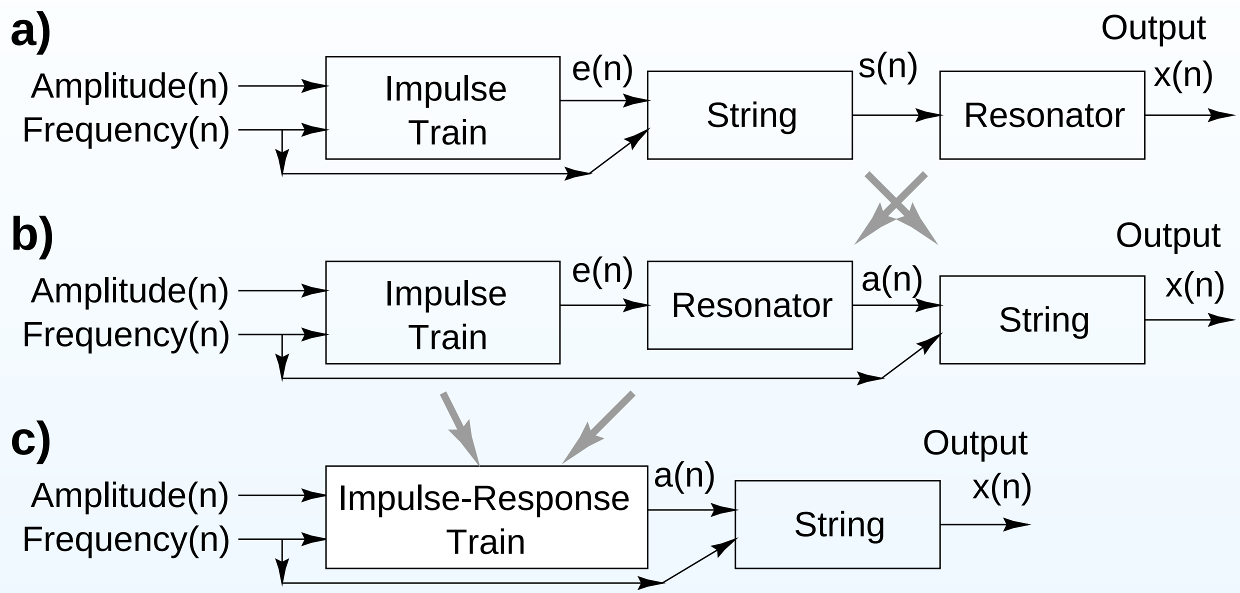
Waveguide Models

Commuted Synthesis
- Acoustic Strings
- Sound Examples
- Linearized Violin
- Commuted Piano
- Pulse Synthesis
- Complete Piano
- Sound Examples

Summary

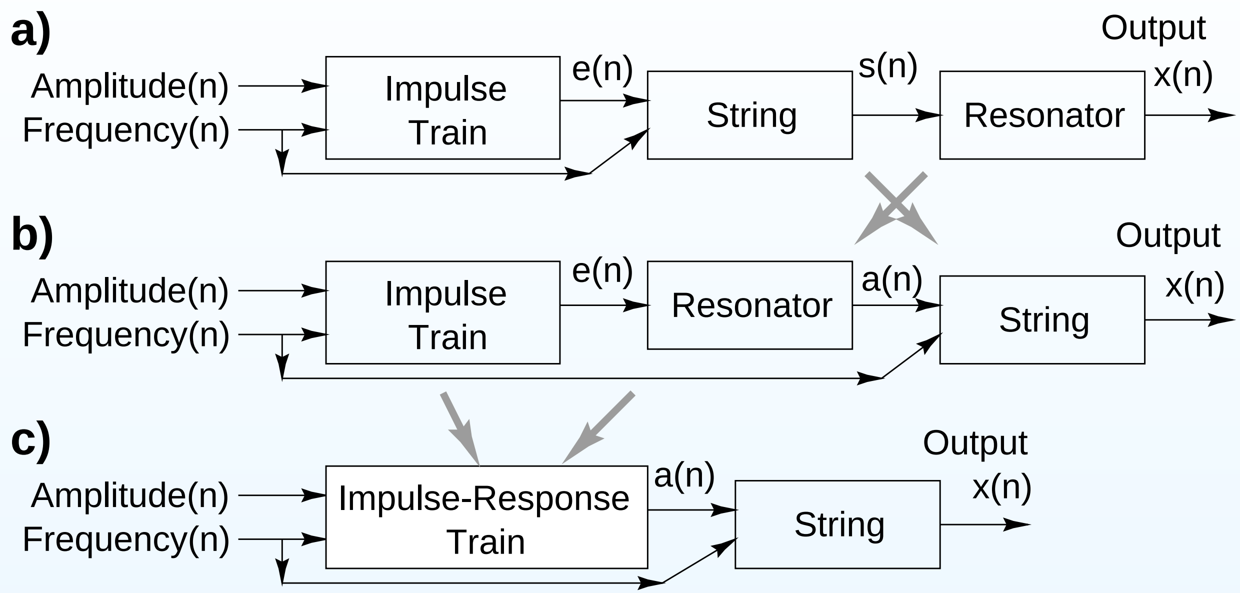
Related Topics

Commuted Synthesis of Linearized Violin



- Assumes *ideal Helmholtz motion* of string
- Sound Examples (Stanford Sondius project, ca. 1995):
  - Bass: (WAV) (MP3)
  - Cello: (WAV) (MP3)
  - Viola 1: (WAV) (MP3)
  - Viola 2: (WAV) (MP3)
  - Violin 1: (WAV) (MP3)
  - Violin 2: (WAV) (MP3)
  - Ensemble: (WAV) (MP3)

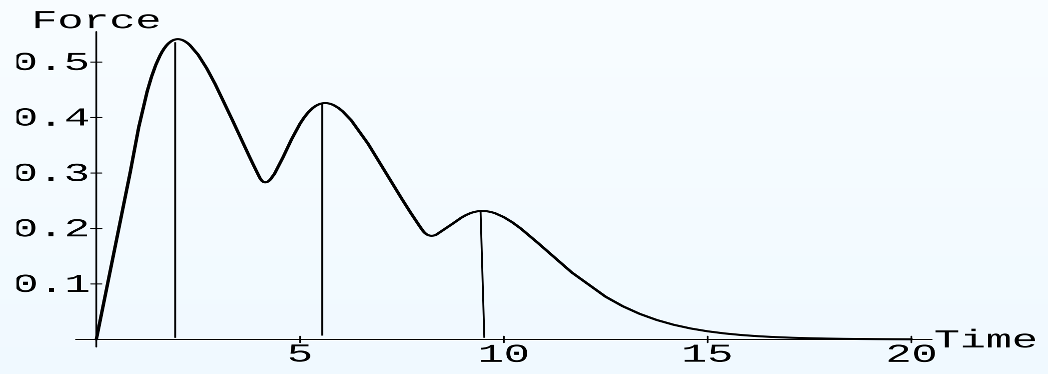
Commuted Synthesis of Linearized Violin



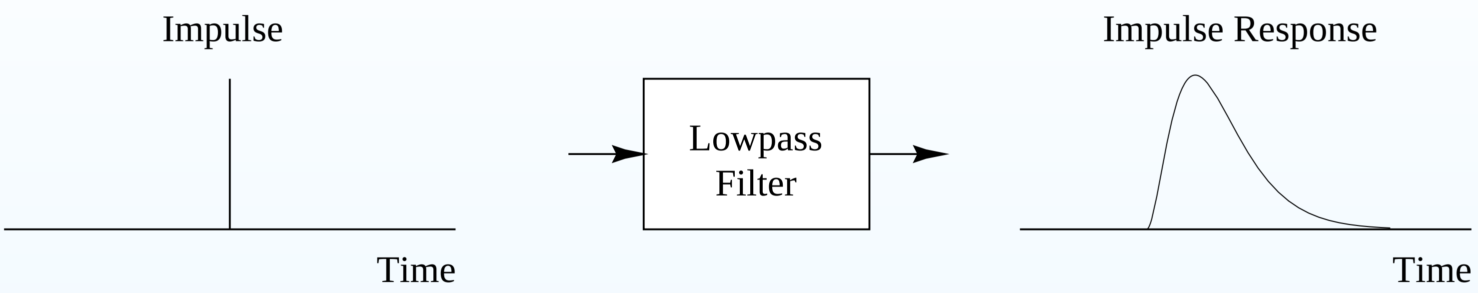
- Assumes *ideal Helmholtz motion* of string
- Sound Examples (Stanford Sondius project, ca. 1995):
  - Bass: (WAV) (MP3)
  - Cello: (WAV) (MP3)
  - Viola 1: (WAV) (MP3)
  - Viola 2: (WAV) (MP3)
  - Violin 1: (WAV) (MP3)
  - Violin 2: (WAV) (MP3)
  - Ensemble: (WAV) (MP3)

Commuted Piano Synthesis (1995)

Hammer-string interaction pulses (force):

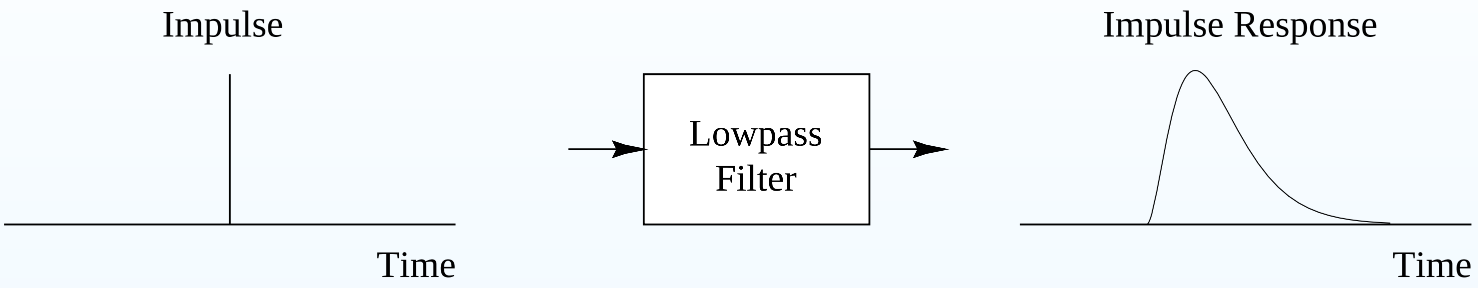


Synthesis of Hammer-String Interaction Pulse



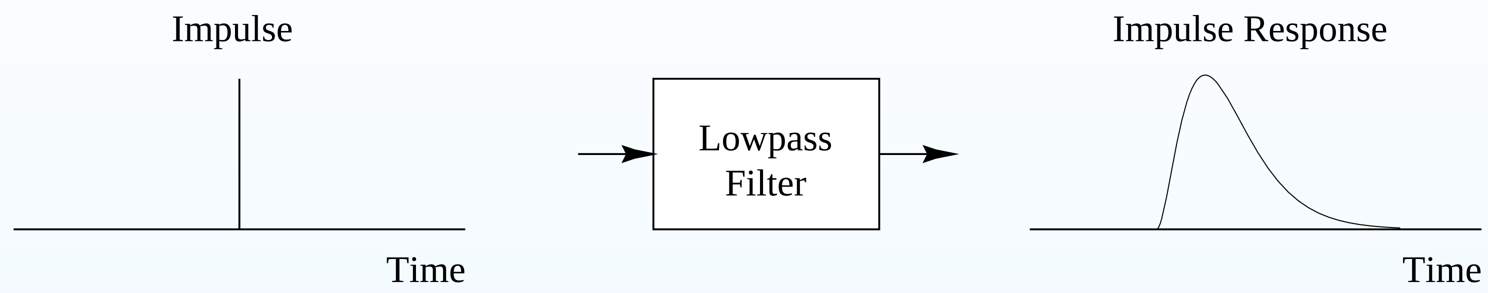
- Faster collisions correspond to *narrower* pulses (*nonlinear filter*)
- For a *given velocity*, filter is linear time-invariant
- Piano is "linearized" for each hammer velocity

Synthesis of Hammer-String Interaction Pulse



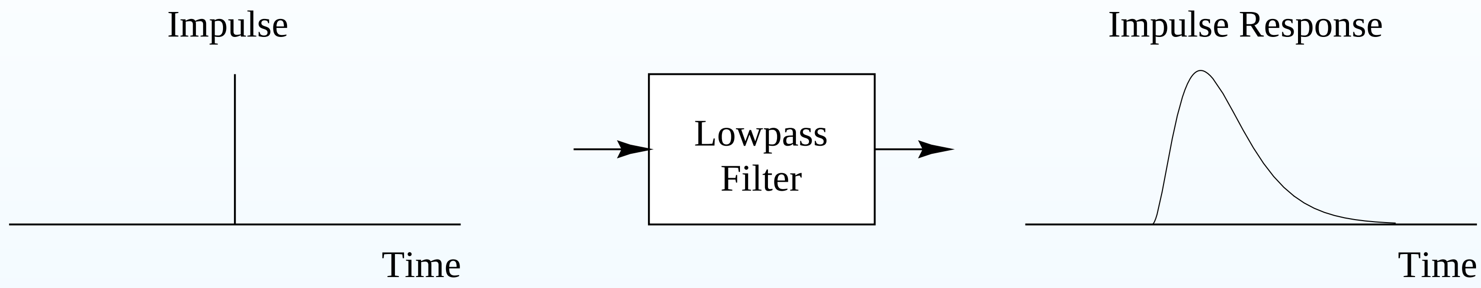
- Faster collisions correspond to *narrower* pulses (*nonlinear filter*)
- For a *given velocity*, filter is linear time-invariant
- Piano is "linearized" for each hammer velocity

Synthesis of Hammer-String Interaction Pulse



- Faster collisions correspond to *narrower* pulses (*nonlinear filter*)
- For a *given velocity*, filter is linear time-invariant
- Piano is "linearized" for each hammer velocity

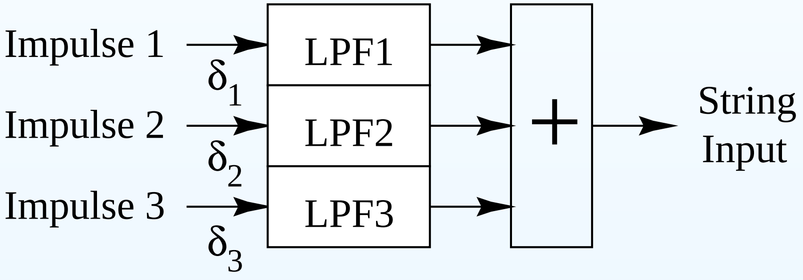
Synthesis of Hammer-String Interaction Pulse

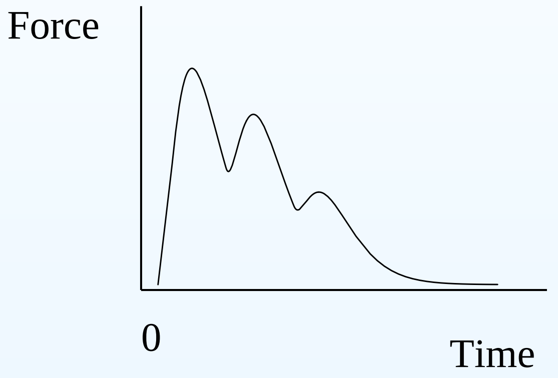


- Faster collisions correspond to *narrower* pulses (*nonlinear filter*)
- For a *given velocity*, filter is linear time-invariant
- Piano is “linearized” for each hammer velocity

Multiple Hammer-String Interaction Pulses

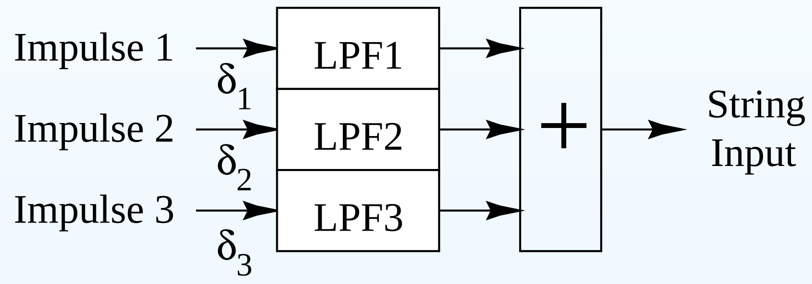
Superimpose several individual pulses:

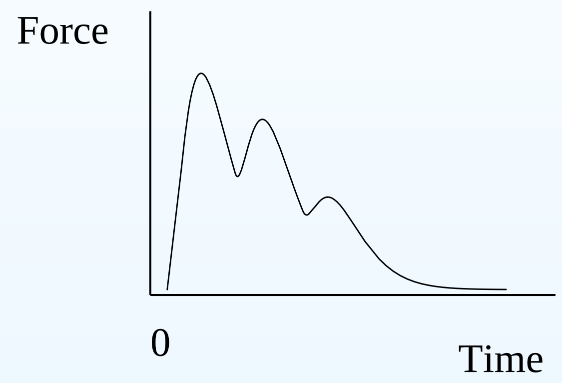




Multiple Hammer-String Interaction Pulses

Superimpose several individual pulses:

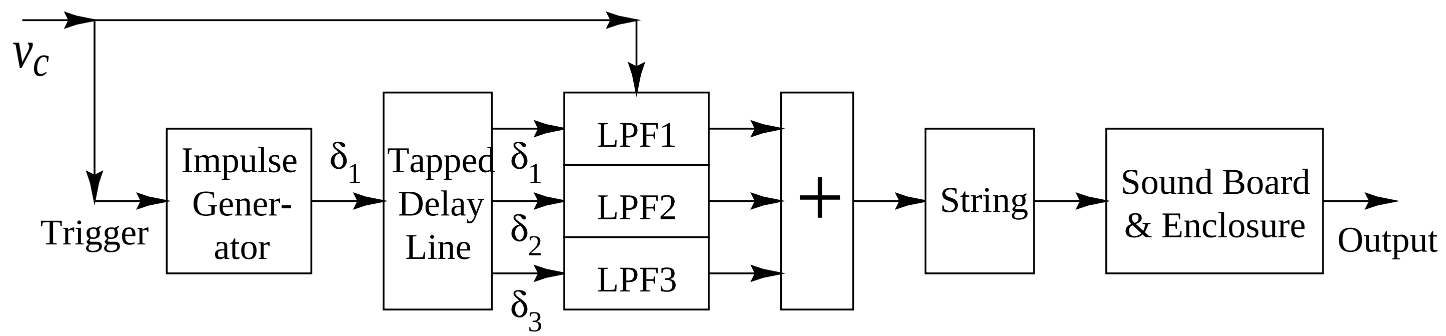




As impulse amplitude grows (faster hammer strike), output pulses become *taller and thinner*, showing less overlap.

Complete Piano Model

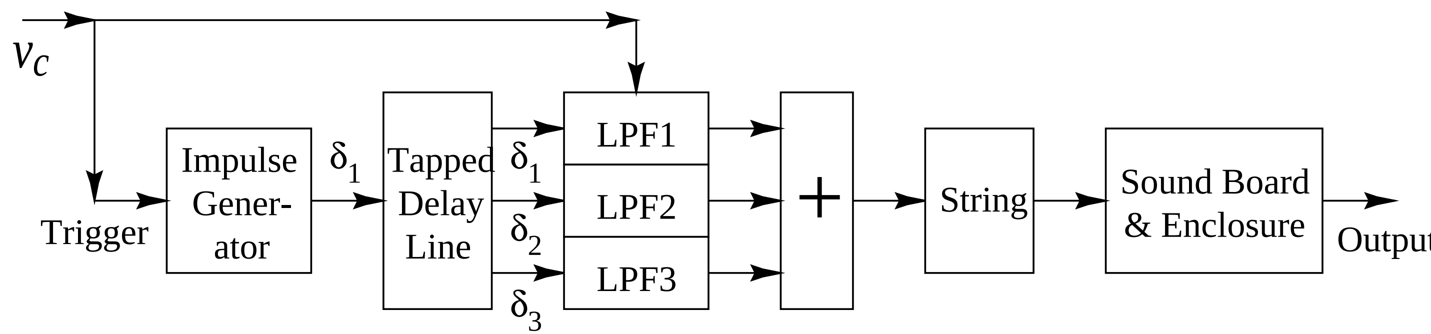
Natural Ordering:



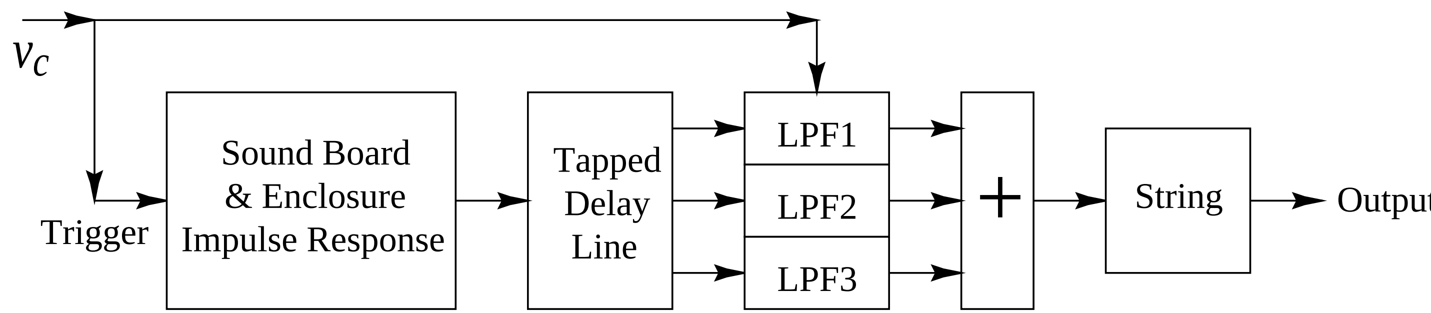
- Soundboard and enclosure are *commuted*
- Only need a stored recording of their *impulse response*
- An enormous digital filter is otherwise required

Complete Piano Model

Natural Ordering:



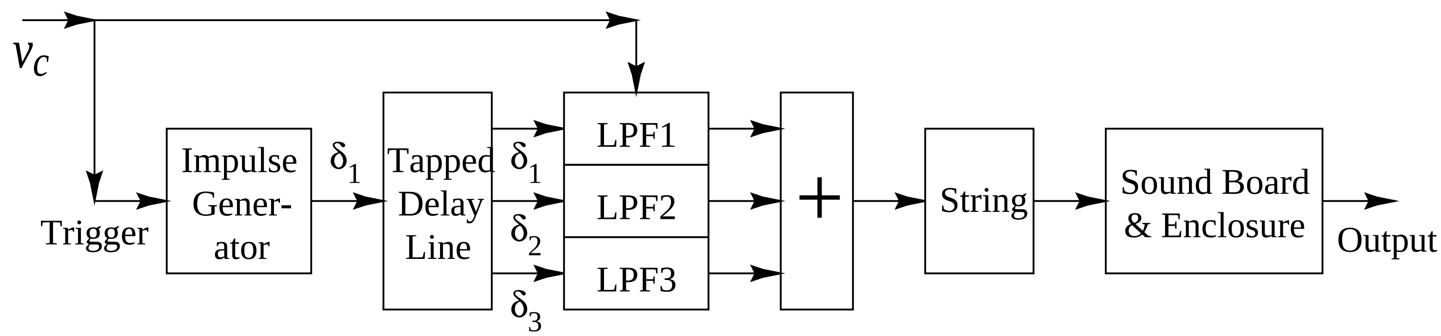
Commuted Ordering:



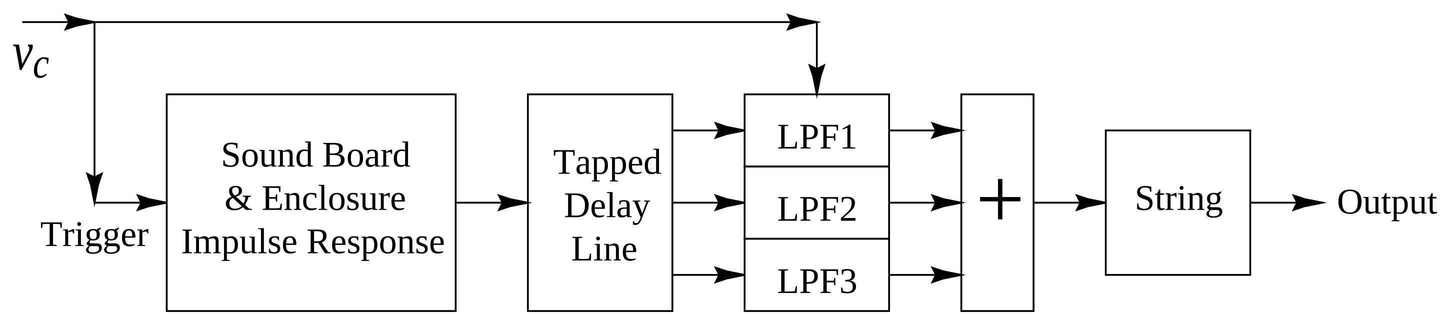
- Soundboard and enclosure are *commuted*
- Only need a stored recording of their *impulse response*
- An enormous digital filter is otherwise required

Complete Piano Model

Natural Ordering:



Commuted Ordering:



- Soundboard and enclosure are *commuted*
- Only need a stored recording of their *impulse response*
- An enormous digital filter is otherwise required

Early DAFx

Delay Effects

Waveguide Models

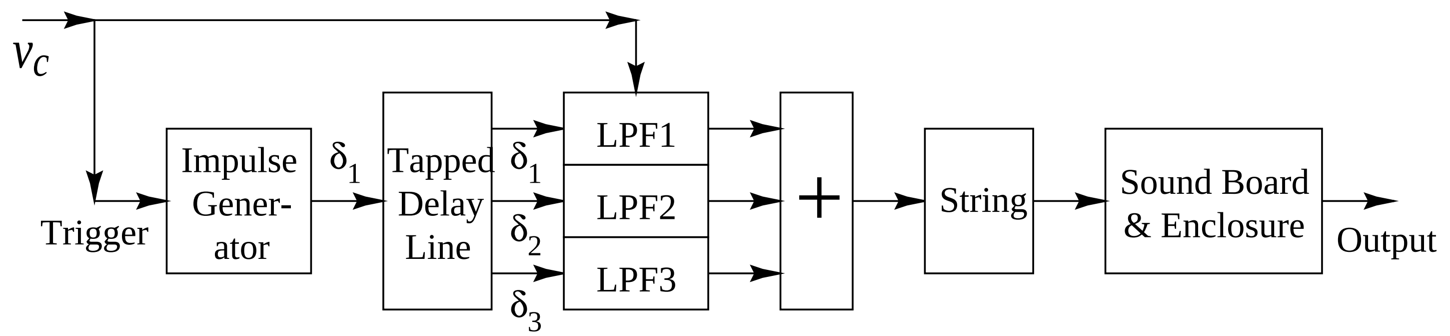
Commuted Synthesis
- Acoustic Strings
- Sound Examples
- Linearized Violin
- Commuted Piano
- Pulse Synthesis
- Complete Piano
- Sound Examples

Summary

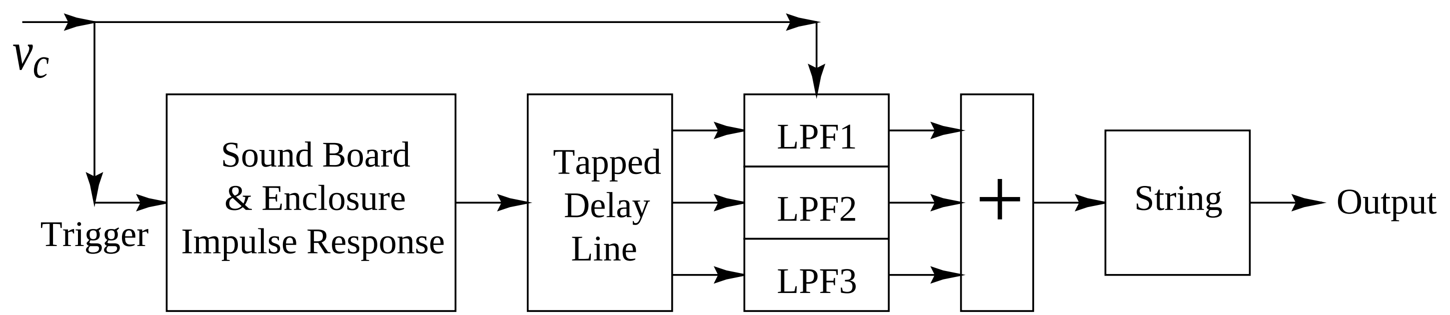
Related Topics

Complete Piano Model

Natural Ordering:



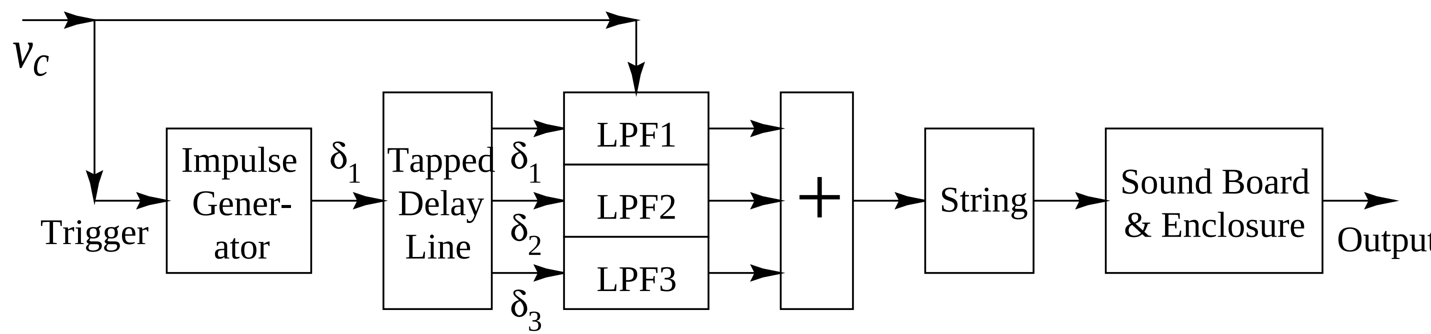
Commuted Ordering:



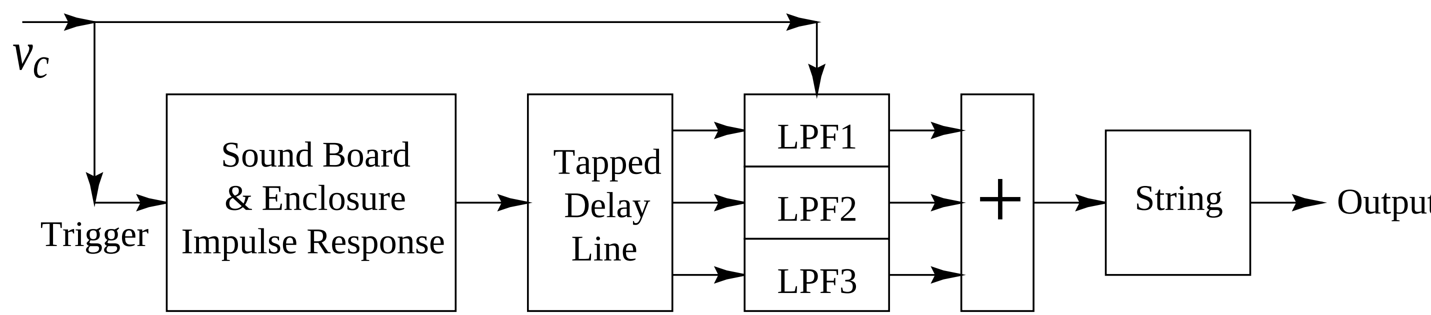
- Soundboard and enclosure are *commuted*
- Only need a stored recording of their *impulse response*
- An enormous digital filter is otherwise required

Complete Piano Model

Natural Ordering:



Commuted Ordering:



- Soundboard and enclosure are *commuted*
- Only need a stored recording of their *impulse response*
- An enormous digital filter is otherwise required

Piano and Harpsichord Sound Examples

(Stanford Sondius Project, ca. 1995)

- Piano: (WAV) (MP3)
- Harpsichord 1: (WAV) (MP3)
- Harpsichord 2: (WAV) (MP3)

More Recent Harpsichord Example

- Harpsichord Soundboard Hammer-Response
- Musical Commuted Harpsichord Example
- More examples

Reference:

"Sound Synthesis of the Harpsichord Using a Computationally Efficient Physical Model",

by Vesa Välimäki, Henri Penttinen, Jonte Knif, Mikael Laurson, and Cumhur Erkut

JASP-2004

`Google` search: *Harpsichord Sound Synthesis*

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Early DAFx

Delay Effects

Waveguide Models

Commuted Synthesis

Summary

Related Topics

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Early DAFx

Delay Effects

Waveguide Models

Commuted Synthesis

Summary

Related Topics

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Early DAFx
Delay Effects
Waveguide Models
Commuted Synthesis
Summary
Related Topics

Summary

We've looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic "rays"
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Early DAFx

Delay Effects

Waveguide Models

Commuted Synthesis

Summary

Related Topics

Summary

We’ve looked at

- Early digital audio effects
- The versatile *filtered delay line*, which efficiently models
  - Acoustic “rays”
  - Strings
  - Bores
  - Horns
  - Any 1D linear wave propagation path (in one direction)
- Digital waveguide models
  - Bidirectional delay lines
  - Scattering at waveguide junctions
  - Nonlinear junctions (for oscillation)
- Commuted synthesis

Early DAFx
Delay Effects
Waveguide Models
Commuted Synthesis
Summary
Related Topics

Related Topics

- Artificial Reverberation
- Virtual Analog
- Related modeling frameworks, such as
  - Finite Difference Schemes
  - Wave Digital Filters
  - Waveguide Mesh

For more, see the online book *Physical Audio Signal Processing* at

`http://ccrma.stanford.edu/~jos/pasp/`