Recent Developments in Musical Sound Synthesis Based on a Physical Model

Julius O. Smith III (jos@ccrma.stanford.edu)
Center for Computer Research in Music and Acoustics (CCRMA)
Department of Music, Stanford University
Stanford, California 94305

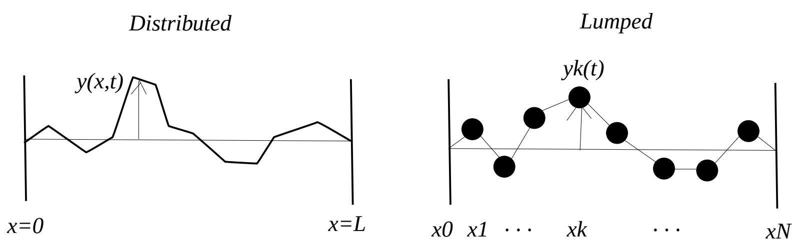
February 19, 2006

Presentation Overheads
Stockholm Musical Acoustics Conference (SMAC-03)
Aug. 6–9, 2003

Outline

- Three Finite Difference Schemes:
  - Differential Equations → Difference Equations
  - Digital Waveguide (DW) Method
  - Wave Digital (WD) Method
- Selected Recent Research
- The Future of Musical Instruments?

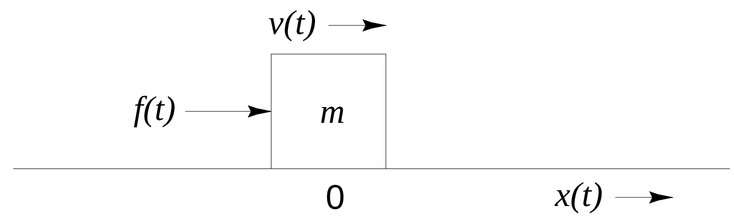
Acoustic Modeling Elements



- Distributed elements (propagating waves):
  - Vibrating strings
  - Woodwinds
  - Brasses
  - Pipes
- Lumped elements
  (vibrating masses, springs, dashpots):
  - Brass-player lip models
  - Woodwind reeds
  - Piano hammer

Finite Difference Approximation (FDA)

Consider the simple differential equation relating velocity and force for an ideal mass:



$$f(t) = m\frac{dv}{dt}$$

Finite Difference Approximation:

$$\begin{aligned} \frac{dv}{dt} &\approx \frac{v_n - v_{n-1}}{T} && \text{(“backward difference”)} \\ &\approx \frac{v_{n+1} - v_{n-1}}{2T} && \text{(“centered difference”)} \end{aligned}$$

E.g.,

$$v_n = v_{n-1} + \frac{T}{m} f_n, \quad n = 0, 1, 2, \ldots .$$

(FDA for a force-driven mass)

Frequency Domain Analysis

FDA, force-drive mass m:

$$v_n = v_{n-1} + \frac{T}{m} f_n, \quad n = 0, 1, 2, \ldots$$

z transform ($v_{-1} = 0$):

$$V(z) = z^{-1} V(z) + \frac{T}{m} F(z)$$

Driving Point Impedance (digital):

$$\boxed{R(z) \triangleq \frac{F(z)}{V(z)} = m \frac{1 - z^{-1}}{T}}$$

Continuous-time driving point impedance:

$$f(t) = m \frac{dv}{dt} \quad \longleftrightarrow \quad F(s) = msV(s)$$

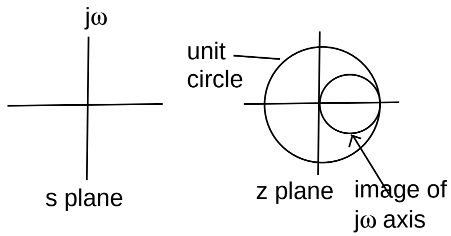
$$\Rightarrow \quad \boxed{R(s) \triangleq \frac{F(s)}{V(s)} = ms}$$

Thus, the FDA maps s plane to the z plane as follows:

$$\boxed{s \leftarrow \frac{1 - z^{-1}}{T}}$$

Properties of Backwards Difference Frequency Mapping

$$s \leftarrow \frac{1 - z^{-1}}{T}$$



- dc ($s = 0$) maps to dc ($z = 1$)
- infinite frequency ($s = \infty$) maps to ($z = 0$)
- no aliasing (mapping is one-to-one)
- frequency axis is *warped*

The continuous and discrete *frequency axes* are related by

$$j\omega_a \leftarrow \frac{1 - e^{-j\omega_d T}}{T} = j\omega_d + \mathcal{O}(\omega_d^2 T)$$

Thus, accurate results can be expected at low frequencies relative to the sampling rate $1/T$.

The Bilinear Transform

A class of *bilinear transforms* map the entire $j\omega$ axis in the s plane exactly once to the unit circle in the z plane:

$$s = c\frac{1 - z^{-1}}{1 + z^{-1}}$$

- dc ($s = 0$) maps to dc ($z = 1$) as for the FDA
- infinite frequency ($s = \infty$) maps to *half the sampling rate* ($z = -1$) instead of $z = 0$ for the FDA
- *damping characteristics are better preserved*
- no aliasing (mapping is one-to-one)
- frequency axis remains warped away from dc

The real constant $c > 0$ allows *one* nonzero frequency (at $s = j\omega_a$) to map exactly to any desired digital frequency (at $z = e^{j\omega_d T}$). All other frequencies are *warped*:

$$j\omega_a = c\frac{1 - e^{-j\omega_d T}}{1 + e^{-j\omega_d T}} = jc\tan\left(\frac{\omega_d T}{2}\right)$$

Bilinear Transform of the Ideal Mass

Starting with the driving point impedance

$$R(s) \stackrel{\Delta}{=} \frac{F(s)}{V(s)} = ms$$

the bilinear transform gives the digital impedance

$$\frac{F_d(z)}{V_d(z)} \stackrel{\Delta}{=} R_d(z) = R\left(c\frac{1 - z^{-1}}{1 + z^{-1}}\right) = mc\frac{1 - z^{-1}}{1 + z^{-1}}$$

Multiplying out

$$F_d(z) + z^{-1}F_d(z) = mcV_d(z) - mcz^{-1}V_d(z)$$

and taking the inverse z transform gives

$$f_n + f_{n-1} = mc\left(v_n - v_{n-1}\right)$$

or

$$\boxed{v_n = v_{n-1} + \frac{1}{mc}\left(f_n + f_{n-1}\right)}$$

(The f_{n-1} term is new relative to the FDA.)

Can check: *Equivalent to trapezoid rule for numerical integration*

Accuracy

Backward-difference approximation:

$$\omega_a \;=\; \omega_d + \mathcal{O}(\omega_d^2 T)$$

Trapezoid rule (bilinear transform):

$$\omega_a \;=\; \omega_d + \mathcal{O}(\omega_d^3 T^2)$$

- Trapezoid rule (bilinear transform) is *second-order* accurate in T.
- Higher order accuracy obtainable using more neighboring grid points.
- How should these extra grid points be brought in?

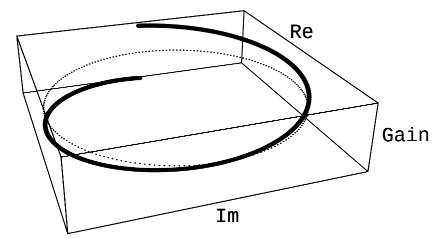
Digital Filter Design Approach

The driving-point impedance $R(s) = ms$ of an ideal mass is an *ideal differentiator* (scaled by m):

$$R(j\omega) = mj\omega.$$

It is therefore natural to define the ideal *digital* differentiator as

$$H(e^{j\omega T}) = j\omega, \quad \omega T \in [-\pi, \pi)$$



- An exact match is *not possible* with a *finite order* digital filter (note frequency-response discontinuity at $z = -1$)
- In practice, we minimize $\left\| H(e^{j\omega T}) - \hat{H}(e^{j\omega T}) \right\|$ where $\hat{H}$ is a *digital filter frequency response*
- We need some *oversampling* in order to have a *guard band* (e.g., from 20 kHz to 22 kHz)
- Desired response is *unconstrained* in the guard band

Limitations of Digital Filter Design

Digital filter design works best for *linear*, *time-invariant* elements.

- What about *nonlinear* systems?
  (Example: clarinet mouthpiece)
- What if physical parameters are *changing* over time?
  (Example: stopped violin string with vibrato)
- What if two different systems *collide*?
  (Example: piano hammer striking a string)

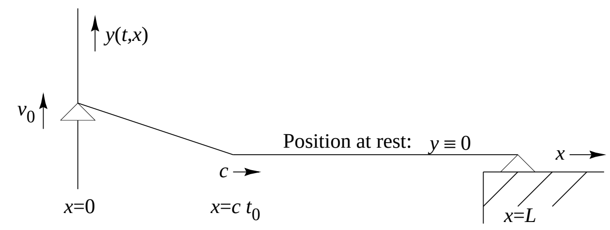
Approaches for nonlinear, time-varying systems:

- Filter *coefficient interpolation*
- Filter *switching*
  (usually *filter state* must be modified as well)
- Filters having a precise physical interpretation
  (true "explicit finite difference schemes")

Both the *digital waveguide* and *wave digital* modeling frameworks have precise physical interpretations. They both therefore classify as physically meaningful explicit finite difference schemes.

Digital Waveguide Modeling

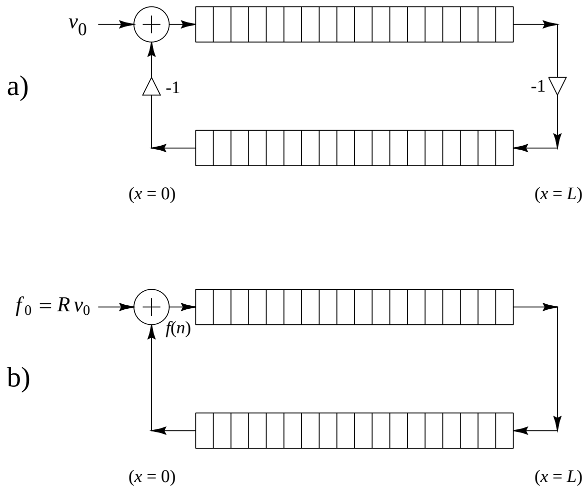
Moving Termination: Ideal String



Moving rigid termination for an ideal string at time $0 < t_0 < L/c$.

- Left endpoint moved at velocity v_0 by an external force $f_0 = Rv_0$, where $R = \sqrt{K\epsilon}$ is the *wave impedance* for transverse waves on the string
- Relevant to *bowed strings* (when bow pulls string)

Waveguide "Equivalent Circuits" for the Uniformly Moving Rigid String Termination

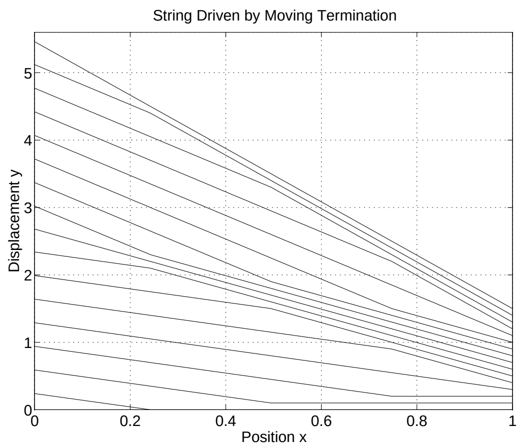


a) Velocity waves.
b) Force waves.

- String moves with speed v_0 or 0 only
- String is always one or two straight segments
- A "Helmholtz corner" (slope discontinuity) shuttles back and forth at speed c
- String slope increases without bound

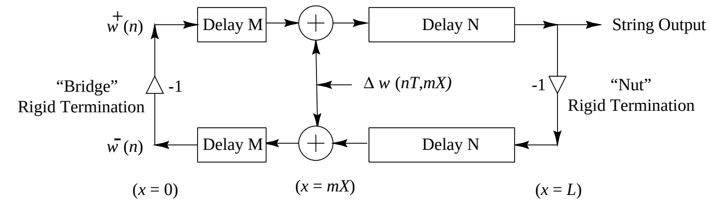
- Applied force at termination steps up to infinity

String Snapshots for Moving Termination

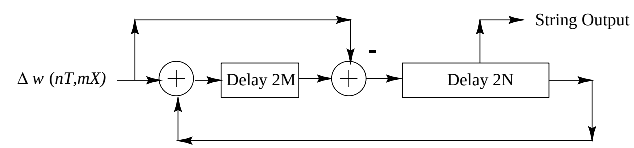


- Successive snapshots of the ideal string with a uniformly moving rigid termination
- Each plot is offset slightly higher for clarity
- GIF89A animation at http://ccrma.stanford.edu/~jos/swgt/movet.html (search jos website for "animation")

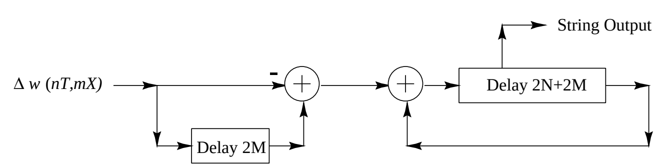
External String Excitation at a Point



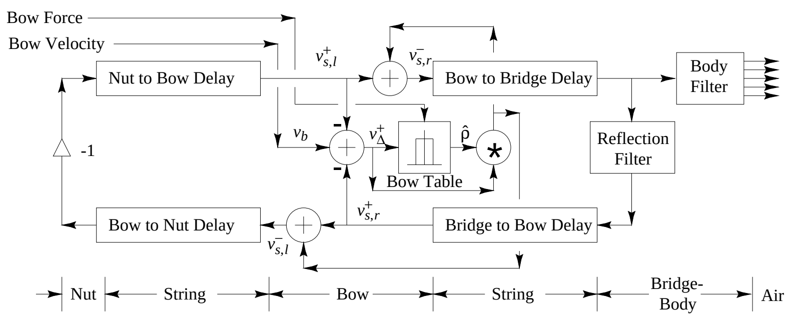
Equivalent System: Delay Consolidation



Equivalent System: Comb Filter Factored Out

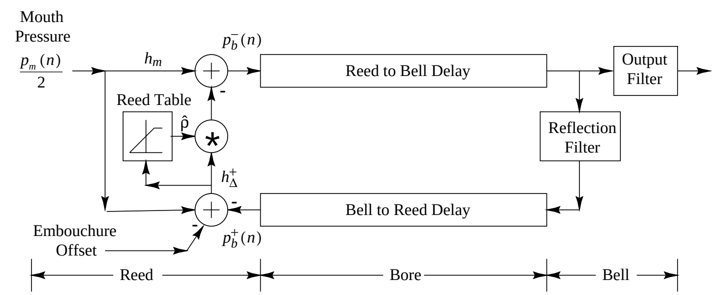


Bowed Strings



- Reflection filter summarizes all losses per period (due to bridge, bow, finger, etc.)
- Bow-string junction = *memoryless* lookup table (or segmented polynomial) ⇒ no thermodynamic model in this version
- Bow-hair dynamics neglected
- Finite bow width neglected

Single-Reed Instruments



Software

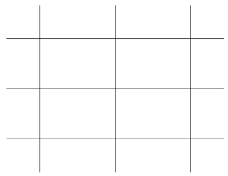
See the Synthesis Toolkit (STK) distributed by CCRMA:
`http://ccrma.stanford.edu/CCRMA/Software/STK/`

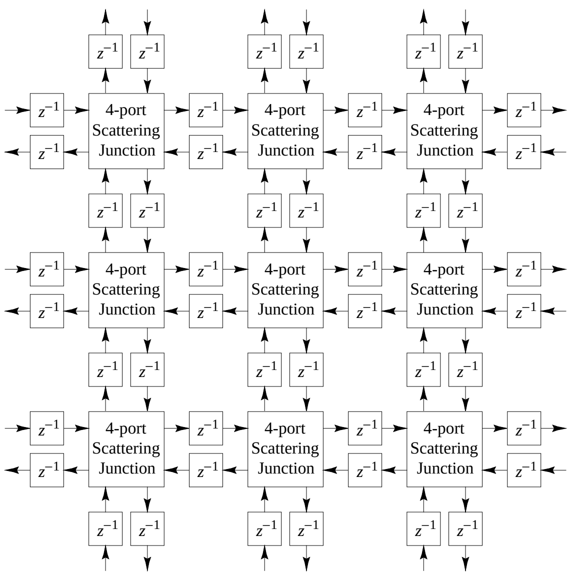
`Google` search: *"STK clarinet"*

Sound Examples

`Google` search: *"waveguide sound examples"*

2D Waveguide Mesh





At each junction:

$$\begin{aligned} V_J &= \frac{\mathsf{in}_1 + \mathsf{in}_2 + \mathsf{in}_3 + \mathsf{in}_4}{2} \\ \mathsf{out}_k &= V_J - \mathsf{in}_k, \qquad k = 1, 2, 3, 4 \end{aligned}$$

Wave Digital Filters

Wave digital elements may be derived as follows:

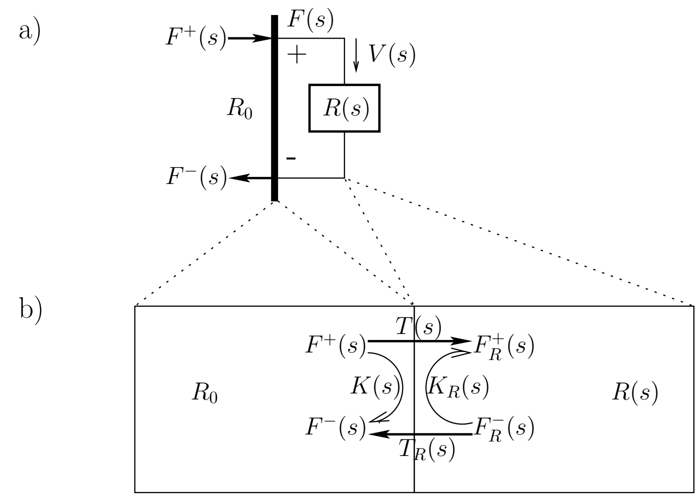
1. Express forces and velocities as *sums of traveling-wave components* (*"wave variables"*):

$$\begin{aligned} f(t) &= f^+(t) + f^-(t) \\ v(t) &= v^+(t) + v^-(t) \end{aligned}$$

   The actual "travel time" is always *zero*.

2. Digitize via the *bilinear transform* (trapezoid rule)
3. Use *scattering junctions* ("*adaptors*") to connect elements together in series and/or parallel.

Physical Construction of Traveling-Wave Element Interfaces



- The inserted waveguide impedance R_0 is *arbitrary* because it was *physically introduced*.
- The element now interfaces to other elements by abutting its waveguide (transmission line) to that of other element(s).
- Such junctions involve *lossless wave scattering*:

$$\begin{aligned} F_R^+(s) &= T(s)F^+(s) + K_R(s)F_R^-(s) \\ F^-(s) &= T_R(s)F_R^-(s) + K(s)F^+(s) \end{aligned}$$

Wave Digital Mass Derivation

For an ideal mass m, we have the *driving point impedance*

$$R(s) = ms$$

which, when used to terminate a waveguide of impedance R_0, gives the reflectance

$$S_m(s) = \frac{ms - R_0}{ms + R_0}$$

(continuous time, Laplace domain). Setting $R_0 = m$ gives

$$S_m(s) = \frac{s-1}{s+1}$$

Digitizing using the bilinear transform gives the digital reflectance

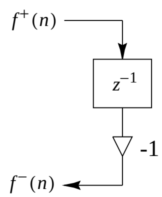
$$\tilde{S}_m(z) \triangleq S_m\left(\frac{1-z^{-1}}{1+z^{-1}}\right) = -z^{-1}$$

The corresponding *difference equation* is then simply

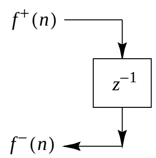
$$\boxed{f^{-}(n) = -f^{+}(n-1)}$$

(wave digital mass).

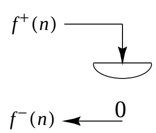
Elementary Wave Flow Diagrams



Wave digital mass



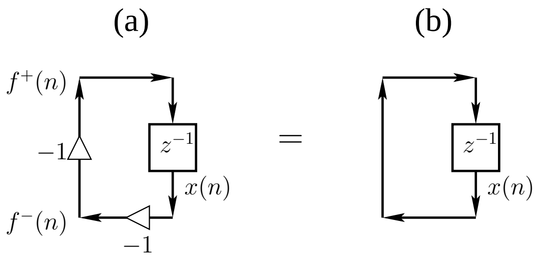
Wave digital spring



Wave digital dashpot

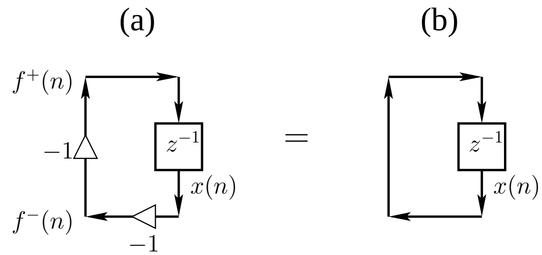
Example: "Piano hammer in flight"

Mass m at constant velocity:



- State variable is in units of *force* $(x(n) \stackrel{\Delta}{=} f^+(n-1))$
- Physical force is $f(n) = f^+(n) + f^-(n) \equiv 0$
- Nonzero state variable $\Rightarrow$ nonzero mass *velocity*

Mass Velocity

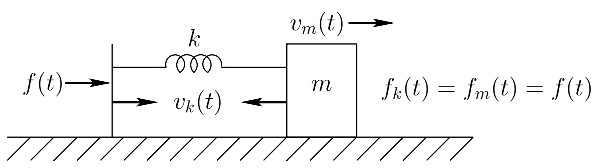


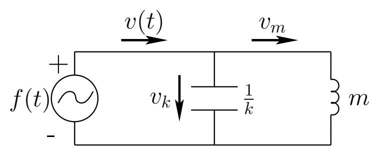
Force-wave simulations easily provide *velocity* outputs:

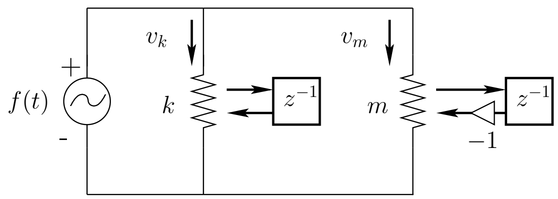
$$\begin{aligned} v(n) &= v^+(n) + v^-(n) = \frac{f^+(n)}{m} - \frac{f^-(n)}{m} \\ &= \frac{x(n)}{m} + \frac{x(n)}{m} = \frac{2}{m} x(n) \end{aligned}$$

Thus, the mass velocity is simply the state variable $x(n)$ scaled by $2/m$.

Spring and Free Mass

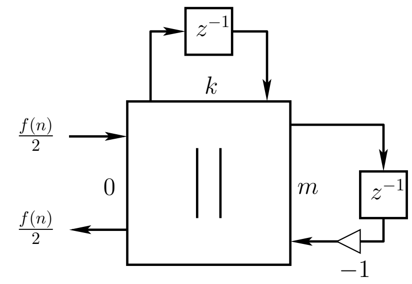




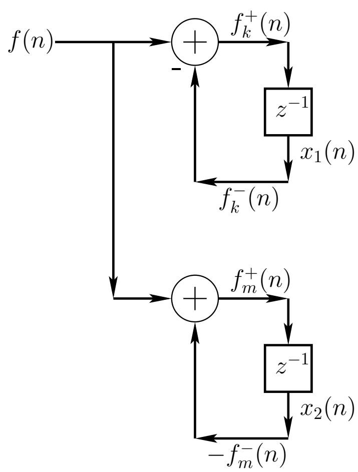


Three parallel branches $\Rightarrow$ *three-port parallel adaptor* needed

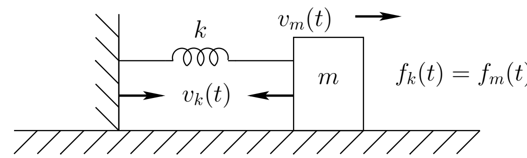
Wave Digital Spring-Mass System

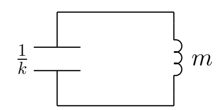


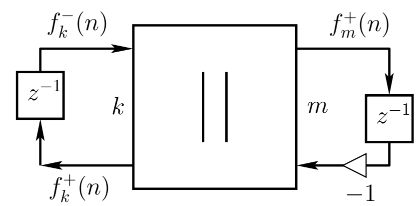
Equivalent Diagram



Wave Digital Mass-Spring Oscillator

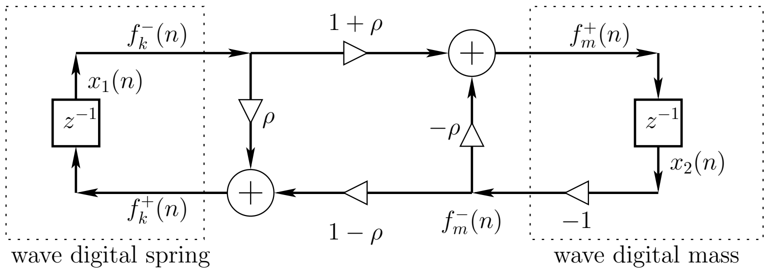






By flipping an element reference direction, we could realize also as a *series* connection.

Expanded Wave Digital Mass-Spring Oscillator



Wave variables to Physical Variables:

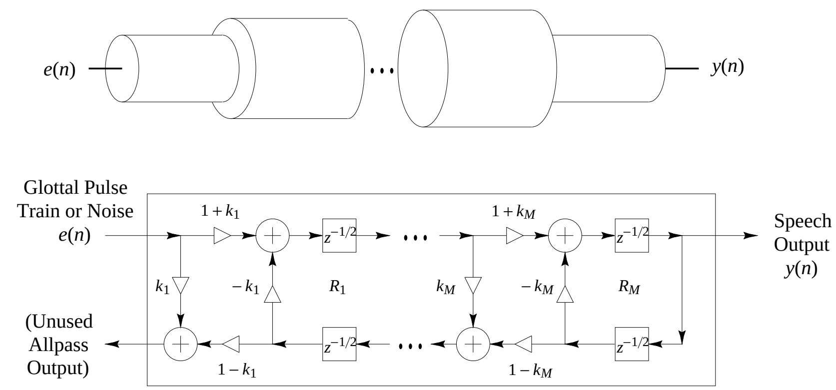
$$\begin{aligned} f_k(n) &= f_k^+(n) + f_k^-(n) \qquad \text{(Spring Force)} \\ f_m(n) &= f_m^+(n) + f_m^-(n) \qquad \text{(Mass Force)} \end{aligned}$$

Reflection coefficient:

$$\rho = \frac{m-k}{m+k}$$

(Impedance step over impedance sum from one infinitesimal waveguide to the next.)

Kelly-Lochbaum Vocal Tract Model



Kelly-Lochbaum Vocal Tract Model (Piecewise Cylindrical)

- Wave impedance in section i:
$$R_i = \frac{\rho c}{A_i}$$
where A_i = cross-sectional area of tube
- Reflection coefficient at ith cylinder-cylinder junction:
$$k_i \triangleq \frac{R_i - R_{i-1}}{R_i + R_{i-1}}$$
- Can be interpreted *either* as a digital waveguide or wave digital model of the vocal tract.
- For correct tuning of two or more resonance (formants), digital waveguide models are required.

Digital Waveguide versus Wave Digital

Digital Waveguides:

- System is *distributed* (vocal tract, bore, string, plate)
- Signals are *traveling waves* digitized via *sampling*
- *Aliasing* occurs for frequencies greater than $f_s/2$
- Frequency axis is *preserved* up to $f_s/2$
- *Relative tuning* is preserved.
- Damping is *preserved*
- Stability preserved

Wave Digital:

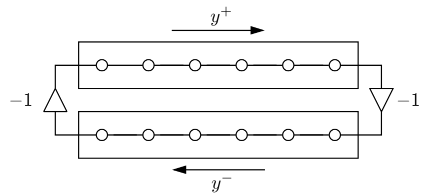
- System is *lumped* (masses, springs, dashpots)
- Signals are *traveling waves* digitized by the *bilinear transform*
- *No aliasing* — entire frequency axis is mapped
- Frequency axis is *nonuniformly warped*
- *One resonance frequency* can be preserved.
- Damping *decreases with frequency*
- Stability preserved

Selected Research Summaries

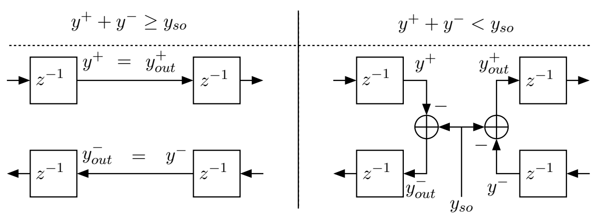
- Colliding Strings
- Feathered Valve Closures
- Joint Vocal-Tract/Glottal-Pulse Model
- Sturm-Liouville Bore Modeling
- Energy Invariant Piano Hammer
- Hyper-Bow meets Virtual Violin

String Collisions

(Krishnaswamy & Smith)

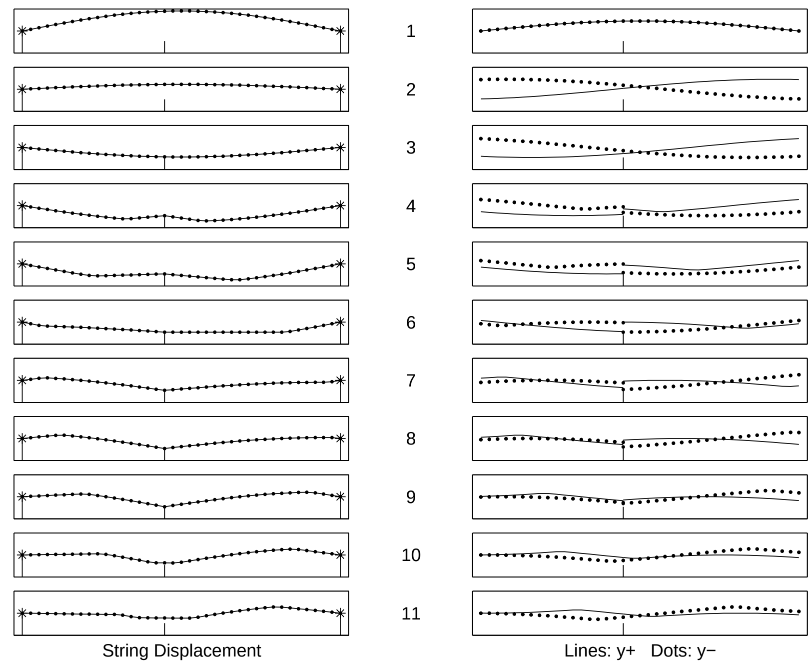


Uniform String

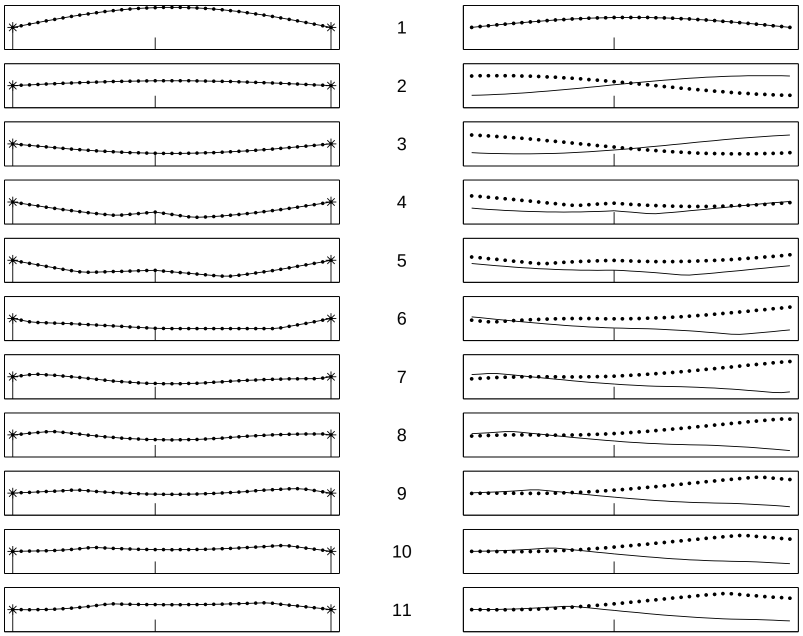


Transition to Collision

Previous Solution



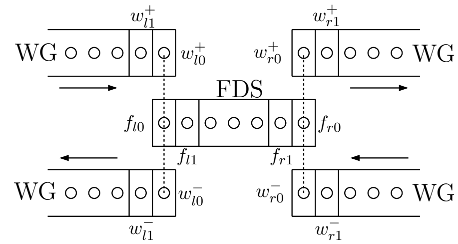
Refined Solution (WASPAA-03)

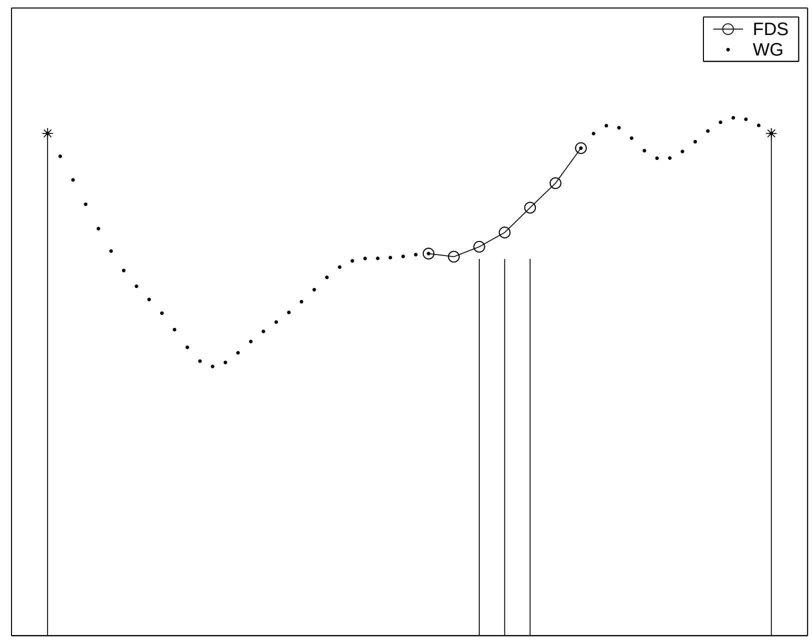


String Displacement

Lines: y+ Dots: y−

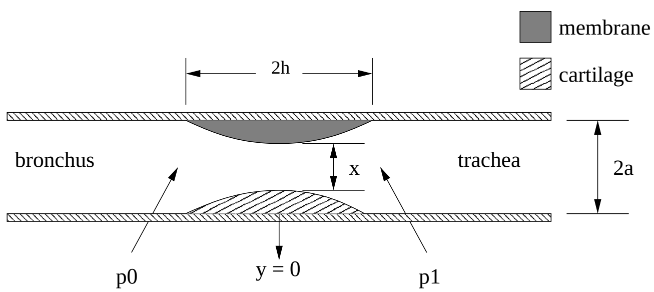
Coupled FDS for Distributed Collision Support





Feathered Valve Closures

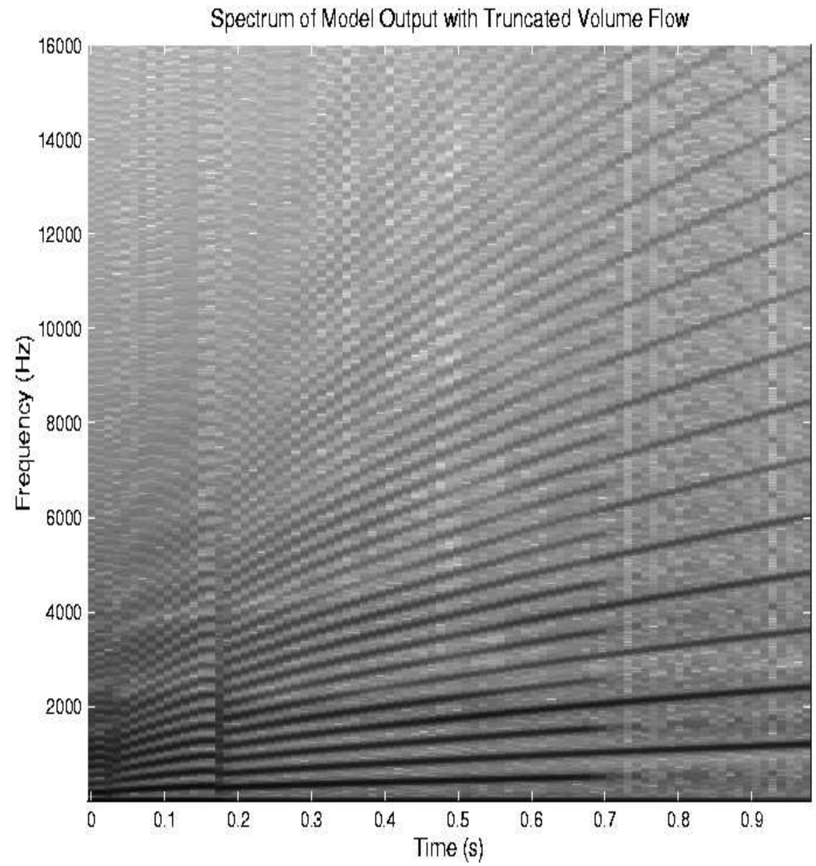
(Smyth, Abel, & Smith)



$$\frac{dU}{dt} = \frac{\sqrt{2}A(t_0)^{\frac{1}{2}}}{\rho}(p_0(t_0) - p_1(t_0)) - \frac{U(t_0)^2}{\sqrt{2}A(t_0)^{\frac{3}{2}}}$$

- The volume-flow Bernoulli term behaves differently depending on whether the channel area $A(t)$ is small or large.
- When $A(t)$ is small and the volume flow $U(t)$ changes direction, the volume flow is truncated to zero.
- **Problem:** Truncation is inaccurate, causing aliasing.

Aliasing During Upward Glissando



Syrinx Spectrogram

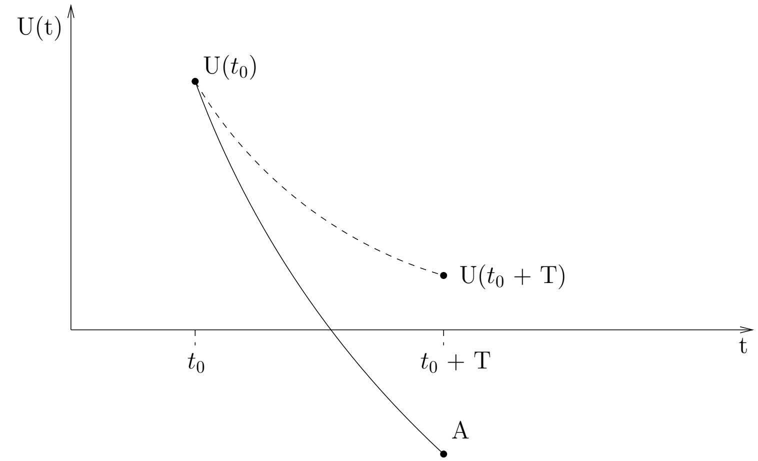
Solution

Solve the differential equation for small channel areas $A(t) \Longrightarrow$

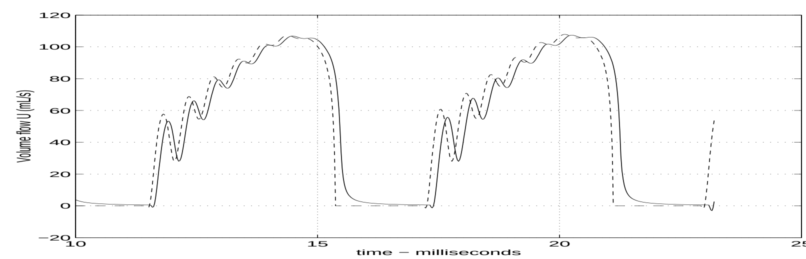
$$\frac{dU}{dt} = \frac{\sqrt{2}A(t_0)^{\frac{1}{2}}}{\rho}[p_0(t_0) - p_1(t_0)] - \frac{U(t_0)^2}{\sqrt{2}A(t_0)^{\frac{3}{2}}} \cdot \frac{1}{1 + \frac{U(t_0)T}{\sqrt{2}A(t_0)^{\frac{3}{2}}}}$$

"Feathering term" $U(t_0)T/\sqrt{2}A(t_0)^{\frac{3}{2}}$ reduces volume flow derivative in the presence of small channel areas $A(t)$ and large sampling periods T, giving a more accurate volume flow and reducing aliasing.

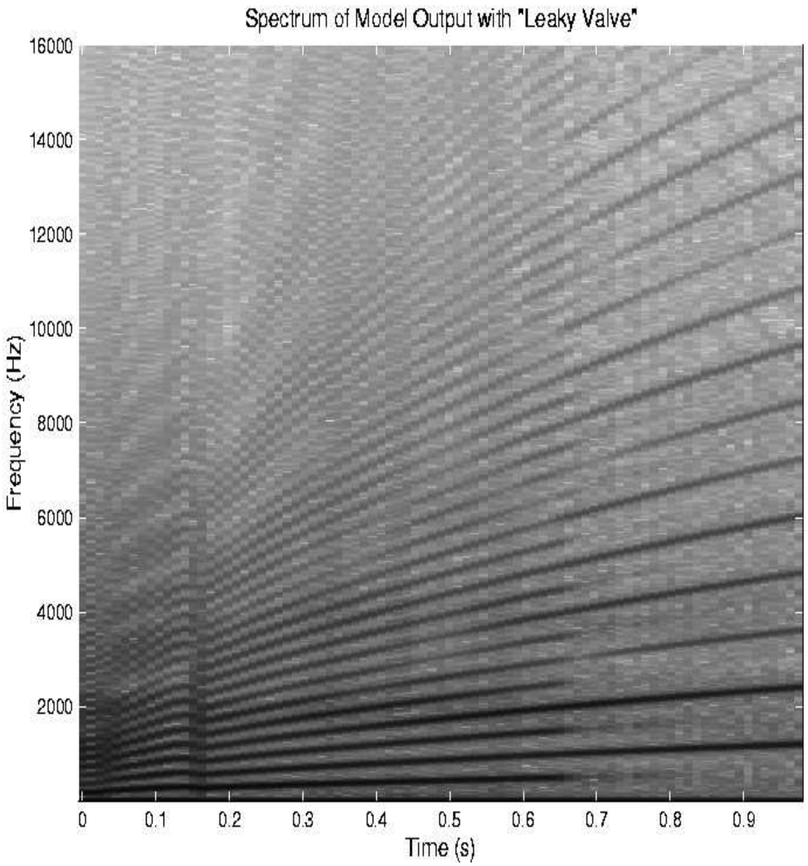
Effect of "Feathering Term" on Volume Flow



Volume Flow vs. Time in Syrinx Oscillation



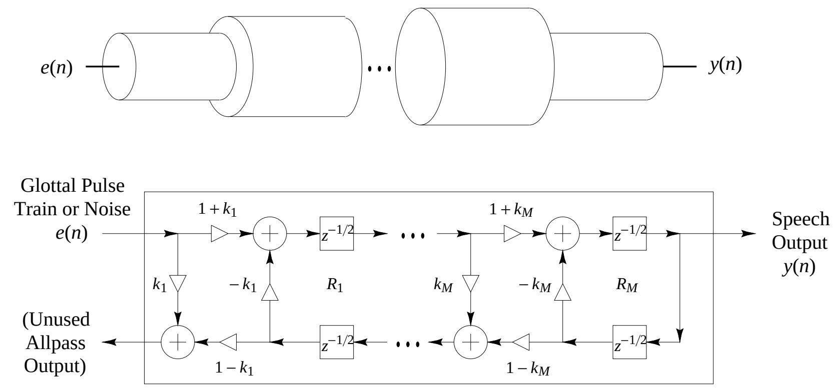
Reduced Aliasing during Upward Glissando



Syrinx Spectrogram

Voice Modeling

Linear Prediction (LP) Vocal Tract Model

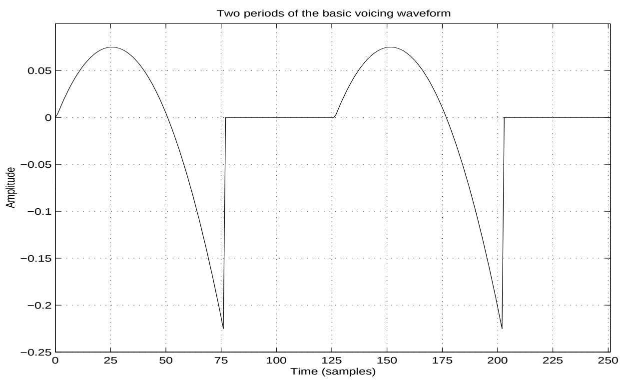


Kelly-Lochbaum Vocal Tract Model (Piecewise Cylindrical)

- Drawn here in Kelly Lochbaum form
- For LP, excitation $e(n)$ must be either an
  - *impulse*, or
  - *white noise*

  This restriction precludes a true physical model
- A *more realistic glottal waveform* $e(n)$ is needed before the vocal tract filter can have the "right shape"

Klatt Derivative Glottal Wave



- Good for estimation:
  - Truncated *parabola* each period
  - Coefficients easily fit to *phase-aligned* inverse-filter output

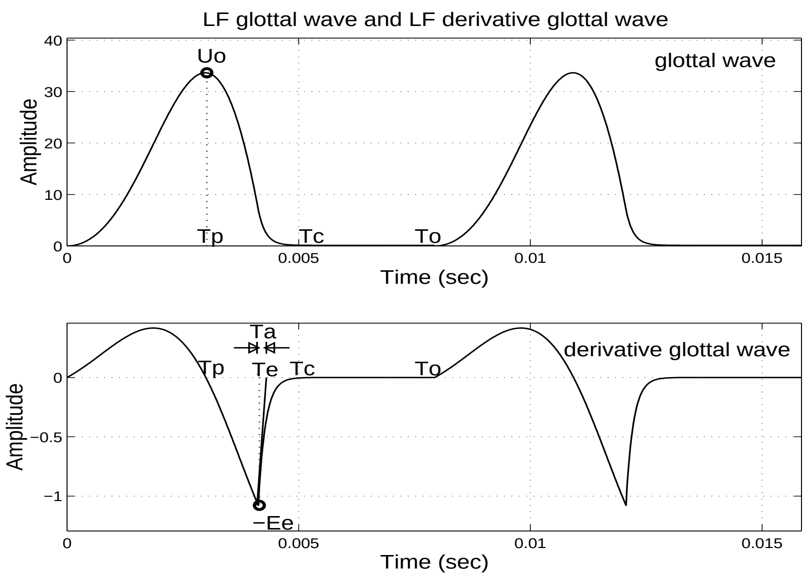
Sequential Unconstrained Minimization

(Hui Ling Lu)
(`Google` search: *"CCRMA Singing/Speech Synthesis"*)

Klatt glottal (parabola) parameters are estimated *jointly* with vocal tract filter coefficients

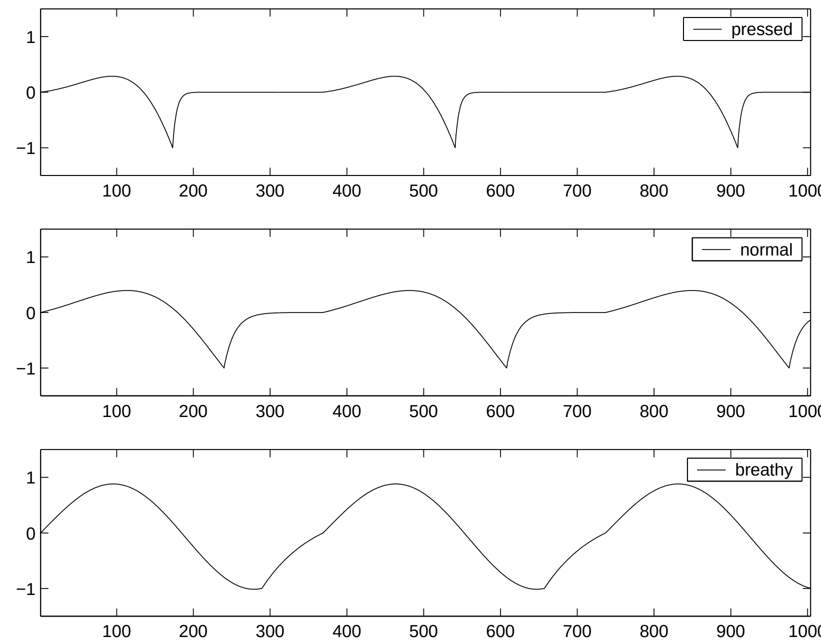
- Formulation resembles that of the *equation error method for system identification*
- For *phase alignement*, we estimate
  - pitch (time varying)
  - glottal closure instant each period
- Optimization is *convex* in all but the phase-alignment dimension

Liljencrantz-Fant Derivative Glottal Wave Model



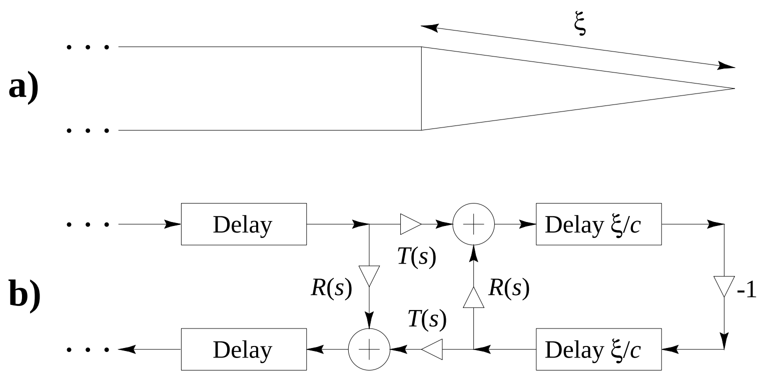
- Better for *intuitively parametrized expressive synthesis*
- LF model parameters are fit to *inverse filter* output
- Use of Klatt model in forming filter estimate yields a "more physical" filter than LP

Parametrized Phonation Types



Conical Digital Waveguide Bore Models

Cylinder with Conical Cap

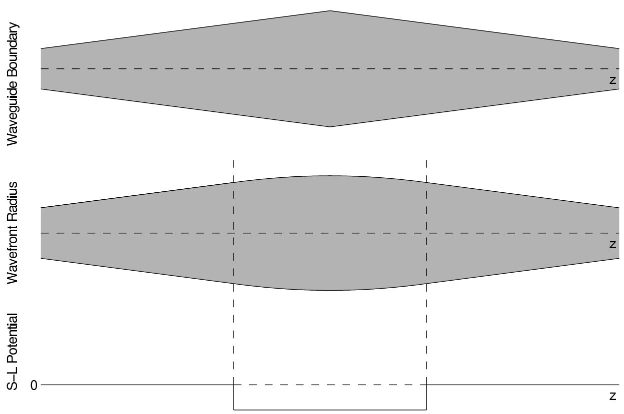


- Cylinder open or closed on left side
- Otherwise closed
- Obviously passive physically
- Reflection filters $R(s)$ and transmission filters $T(s)$ are *unstable one-pole filters*.
- Instability is "canceled" by reflection from tip. (More precisely, there is no instability.)

Sturm-Liouville Scattering

(David P. Berners)
(`Google` search: *"David Berners"*)

Instead of an abrupt change, spread it out over a *region*:



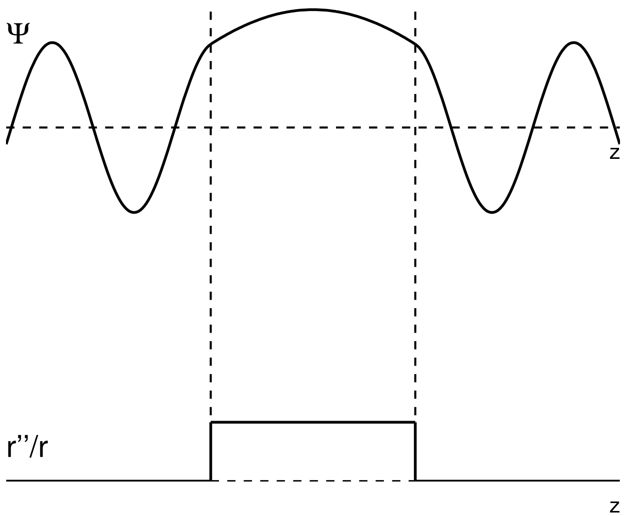
(From David Berners EE/CCRMA thesis)

- Waveguide Curvature $\leftrightarrow$ *Potential Function*:

$$V(x) \propto \frac{r''(x)}{r(x)}$$

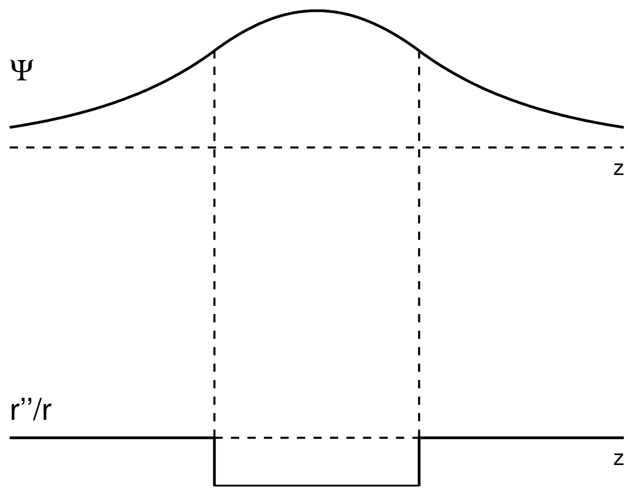
where $r(x)$ is the radius of the *wavefront* as a function of position x along the cone axis.

Traveling Mode through a Potential Barrier



(From David Berners EE/CCRMA thesis)

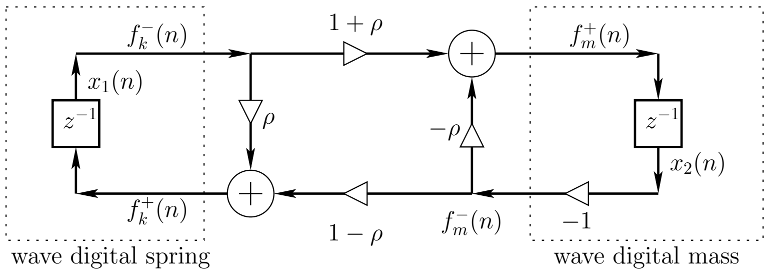
Trapped Mode in a Potential Well



(From David Berners EE/CCRMA thesis)

Normalized Wave Digital Piano Hammer

Wave Digital Mass-Spring Oscillator



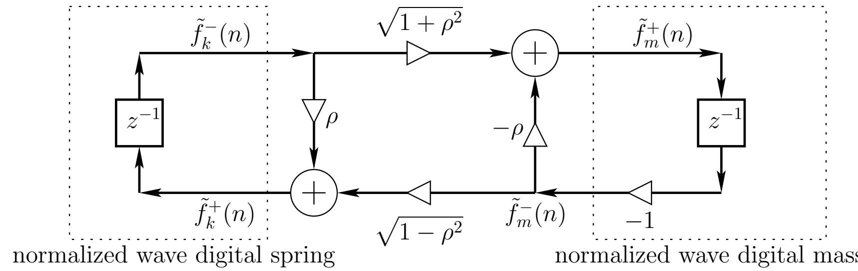
Convert force waves to root-power waves:

$$\tilde{f}_i^+ \triangleq \frac{f_i^+}{\sqrt{R_i}}$$

$$\tilde{v}_i^+ \triangleq v_i^+ \cdot \sqrt{R_i}$$

where R_i = wave impedance in waveguide i.

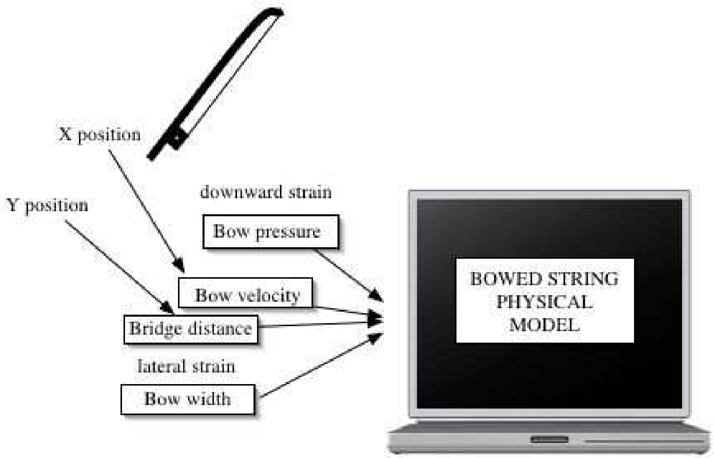
Normalized Wave Digital Mass-Spring Oscillator



- Stored energy *invariant* with respect to *time varying* and/or *nonlinear* changes in mass or spring constants
- For the piano hammer, the felt spring constant may vary with force $f_k(n)$ without altering stored energy
- Only the reflection coefficient $\rho(n)$ varies as the spring is compressed
- Note delay-free interdependence of $\rho(n)$ and $f_k(n)$
- See Bensa & Bilbao et al. (SMAC-03) for an update on the *lossy* normalized wave digital piano hammer

The Virtual Violin Project

(Serafin & Young)



The goal of this project is to create a virtual bowed string instrument that

- reproduces traditional bow strokes known to players,
- extends possibilities offered by traditional instruments.

We use a bow with a wireless sensing system connected to a real-time waveguide bowed string physical model.

The Future of Musical Instruments?

Some Observations

- In the beginning of computer music, there was the *tape piece*, usually for four channels.
- The audience sat in a darkened auditorium with nothing to look at but the speakers.
- Since then, there has been increasing integration of *live performance*.
- Initially, performers were *slaves* to the computer clock.
- Nowadays, computers are "listening" more, allowing performers more control.

Speculations

- Extrapolating, we might predict that live performances will be "technology enhanced" in increasingly subtle ways.
- There is always room for new *controller interfaces*, enabling new kinds of live performance.
- Audio and visual experiences can be much more tightly integrated than is typical now.
- "Virtual Environments" will become more complete, compelling, and interactive.
- Traditional musical instruments could evolve into *controllers* for *virtual* musical instruments.
  - The Serafin & Young Virtual Violin Project is one example
  - Controllers need not be responsible for the final sound

Related Fronts

- Virtual Performers
- Synthesis-Based Audio Coders
  - MPEG-4 (SAOL)
  - Noise reduction by resynthesis
- Speaker Arrays