CENTER FOR COMPUTER RESEARCH IN MUSIC AND ACOUSTICS
DECEMBER 1990

Department of Music
Report No. STAN-M-67

Efficient Yet Accurate Models for Strings and Air Columns using Sparse Lumping of Distributed Losses and Dispersion

Julius Orion Smith III

CCRMA
DEPARTMENT OF MUSIC
Stanford University
Stanford, California 94305

Efficient Yet Accurate Models for Strings and Air Columns using Sparse Lumping of Distributed Losses and Dispersion

Julius O. Smith III

Assoc. Prof. (Research), CCRMA, Music Dept., Stanford University, Stanford, CA 94305
Signal Processing Engineer, NeXT Inc., 900 Chesapeake Dr., Redwood City, CA 94063
email: jos@next.com or jos@ccrma.stanford.edu

Colloquium on Physical Modeling
Grenoble 1990

Introduction

Ever since Newton discovered his famous laws of motion, physicists have been applying them to natural phenomena, deriving differential equations which govern their motion. Analytic solutions have given much insight into the dynamic behavior of physical systems. In more recent history, differential equations have provided a direct basis for computational modeling. They can be numerically integrated by computer to simulate the response of the physical system to external forces and boundary conditions. Because the mathematical description is local, giving relations at only a single point in time and space, numerical integration is normally very expensive as a computational modeling technique.

There are situations, however, in which a computational solution based on differential equations is highly economical. This happens with musical instrument models because systems constructed to produce musical sound have two important properties: (1) Losses are small, compared with a period of sound, and (2) phase is weakly perceived on a time scale of one period. These properties enable the following solution approach:

- Solve the wave equation analytically in the free medium, with no boundary conditions.
- Simulate the solution directly (traveling waves) instead of numerically integrating.
- Commute losses and dispersion to the medium boundaries, consolidating them.
- Simulate the consolidated losses and dispersion using a single, low-order, digital filter.

Interior to the medium, away from inputs, outputs, and boundary conditions, we simulate a lossless idealization where a superposition of traveling waves may be implemented using pure delays. This can result in a huge reduction in the computational burden without affecting simulation accuracy.

The Lossless One-Dimensional Wave Equation

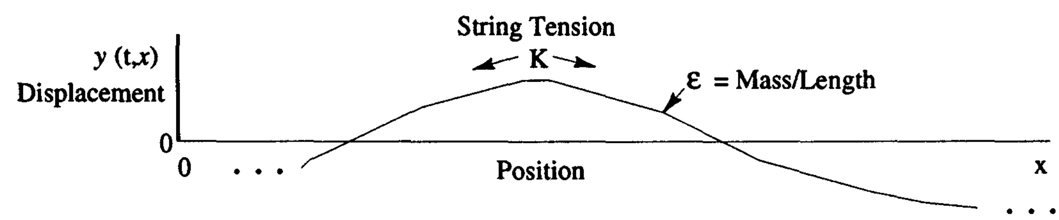

Figure 1. The ideal vibrating string.

Newton's law equating force to mass times acceleration leads to the following *wave equation* for transverse waves along the ideal string, depicted in Figure 1 (Morse 1936):

$$Ky'' = \epsilon\ddot{y}$$

where

$$K \triangleq \text{string tension} \qquad y \triangleq y(t,x)$$

$$\epsilon \triangleq \text{linear mass density} \qquad \dot{y} \triangleq \frac{\partial}{\partial t}y(t,x)$$

$$y \triangleq \text{string displacement} \qquad y' \triangleq \frac{\partial}{\partial x}y(t,x)$$

The same equation applies to any perfectly elastic medium which is displaced along one dimension, such as a column of air in a clarinet or organ pipe. We refer to the general class of such media as *one-dimensional waveguides.*

Solution

Solutions to the wave equation can be obtained by numerically integrating the differential equations, and string sounds have been computed based on such an approach (Ruiz 1969, Hiller and Ruiz 1971). Before the availability computers, however, the wave equation was typically solved in closed form by finding analytic functions which satisfy the differential equations. Because e^{st} is an *eigenfunction* under differentiation (i.e., the exponential function is its own derivative), it is normally profitable to plug in a generalized exponential function, with maximum degrees of freedom in its parametrization, to see if parameters can be found to fulfill the constraints imposed by the equations.

In the case of the one-dimensional wave equation with no boundary conditions, an appropriate choice of eigensolution is

$$y(t,x) = e^{st+vx}$$

Substituting into the wave equation yields

$$\begin{array}{ll} \dot{y} = sy & y' = vy \\ \ddot{y} = s^2 y & y'' = v^2 y \end{array}$$

Defining *wave velocity* as $c \triangleq s/v$, the wave equation becomes

$$Kv^2 y = \epsilon s^2 y \;\Rightarrow\; \frac{K}{\epsilon} = \frac{s^2}{v^2} \triangleq c^2$$

$$\Rightarrow\; v = \pm\frac{s}{c}$$

Thus

$$y(t,x) = e^{s(t \pm x/c)}$$

is a solution for all s. By *superposition,*

$$y(t,x) = \sum_s A^+(s)e^{s(t-x/c)} + A^-(s)e^{s(t+x/c)}$$

is also a solution, where $A^+(s)$ and $A^-(s)$ are arbitrary complex-valued functions of s. Setting $s \triangleq jw$, we have, by Fourier's theorem,

$$y(t,x) = f\left(t - \frac{x}{c}\right) + g\left(t + \frac{x}{c}\right)$$

for *arbitrary* continuous functions $f(\cdot)$ and $g(\cdot)$. This is the traveling-wave solution of the wave equation attributed to D'Alembert. A function of $(t - x/c)$ may be interpreted as a fixed waveshape traveling to the *right,* (i.e., in the *positive* x direction), with speed c. Similarly, a function of $(t + x/c)$ may be seen as a wave traveling to the *left* (negative x direction) at c meters per second.

Digital Simulation

To simulate the solution of the ideal, one-dimensional wave equation, we simply *sample* the traveling-wave amplitude at intervals of T seconds, which implies a *spatial* sampling interval of $X \triangleq cT$ meters. Sampling is carried out by the change of variables

$$\begin{aligned} x &\to x_m = mX \\ t &\to t_n = nT \end{aligned}$$

and this substitution gives

$$\begin{aligned} y(t_n, x_m) &= f(t_n - x_m/c) + g(t_n + x_m/c) \\ &= f(nT - mX/c) + g(nT + mX/c) \\ &= f\left[(n-m)T\right] + g\left[(n+m)T\right] \end{aligned}$$

Define

$$y^+(n) \triangleq f(nT) \qquad\qquad y^-(n) \triangleq g(nT)$$

Then a section of the ideal string or waveguide can be simulated as shown in Figure 2 (where outputs have been arbitrarily placed at $x = 0$ and $x = 3X$).

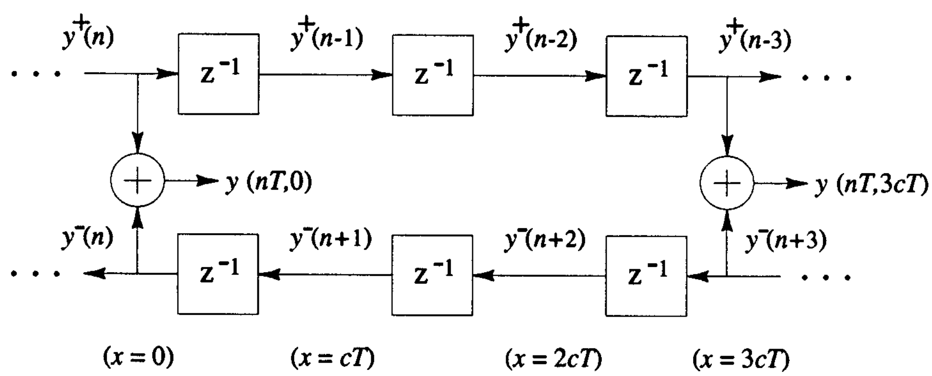

Figure 2. Digital simulation of the ideal, lossless waveguide with observation points at $x = 0$ and $x = 3X = 3cT$.

A more compact simulation diagram which stands for either sampled or continuous simulation is shown in Figure 3. The figure emphasizes that the ideal, lossless waveguide is simulated by a *bidirectional delay line*.

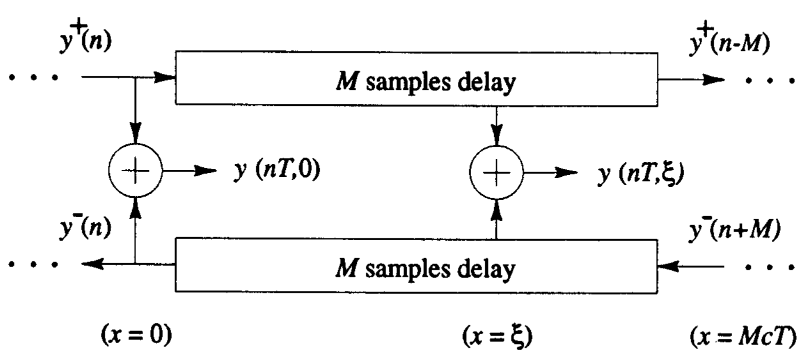

Figure 3. Simplified picture of ideal waveguide simulation.

It is important to note that the digital simulation of the traveling-wave solution to the lossless, one-dimensional wave equation (e.g., the ideal string) is *exact* at the sampling instants, (to within the numerical precision), provided the waveshapes traveling along the string are initially *bandlimited* to less than half the sampling frequency. In other words, the highest frequencies present in the signals $f(t)$ and $g(t)$ may not exceed half the temporal sampling frequency $f_s \triangleq 1/T$; equivalently, the highest *spatial* frequencies in the shapes $f(x/c)$ and $g(x/c)$ may not exceed half the spatial sampling frequency $\nu_s \triangleq 1/X$.

Bandlimited *spatial* interpolation may be used to construct an output for an arbitrary x not a multiple of cT, as suggested by the output drawn in Fig. 3. Similarly, bandlimited interpolation across time (Smith and Gossett 1984) serves to evaluate the waveform at an arbitrary time not an integer multiple of T. Because bandlimited interpolation is always approximate in practice, being primarily limited by the quality of the lowpass filter used in the interpolation, the signal at interpolated times and positions will not be exact to within numerical precision, except perhaps within a frequency band strictly less than (e.g., 90% of) $f_s/2$.

The Lossy One-Dimensional Wave Equation

In any real vibrating string, there are energy losses due to yielding terminations, drag by the surrounding air, and internal friction within the string. A wide class of losses, such as the wave impedance and viscosity of air, exert a purely resistive force proportional to transverse velocity of the string. If this proportionality constant is μ, we obtain the modified wave equation

$$Ky'' = \epsilon\ddot{y} + \mu\dot{y}$$

Again let $y(t,x) = e^{st+vx}$ to find the relationship between temporal and spatial frequencies in the eigensolution, and use Fourier superposition to build the general class of solutions. The wave equation becomes

$$\begin{aligned}
K\left(v^2 y\right) &= \epsilon\left(s^2 y\right) + \mu\left(sy\right) \;\Rightarrow\; Kv^2 = \epsilon s^2 + \mu s \\
\Rightarrow\; v^2 &= \frac{\epsilon}{K}s^2 + \frac{\mu}{K}s = \frac{\epsilon}{K}s^2\left(1+\frac{\mu}{\epsilon s}\right) \triangleq \frac{s^2}{c_0^2}\left(1+\frac{\mu}{\epsilon s}\right) \\
\Rightarrow\; v &= \pm\frac{s}{c_0}\sqrt{1+\frac{\mu}{\epsilon s}}
\end{aligned}$$

where $c_0 \triangleq \sqrt{K/\epsilon}$ is the wave velocity in the lossless case. At high frequencies (large $|s|$), or when the friction coefficient μ is small relative to the mass density ϵ, we have the approximation

$$\left(1+\frac{\mu}{\epsilon s}\right)^{\frac{1}{2}} \approx 1 + \frac{1}{2}\frac{\mu}{\epsilon s}$$

by the binomial theorem. For this small-loss approximation, we obtain the following relationship between temporal and spatial frequency:

$$v \approx \pm\frac{1}{c_0}\left(s+\frac{\mu}{2\epsilon}\right)$$

The eigensolution is then

$$e^{st+vx} = \exp\left[st \pm \left(s+\frac{\mu}{2\epsilon}\right)\frac{x}{c}\right] = \exp\left[s\left(t\pm\frac{x}{c}\right)\right]\exp\left(\pm\frac{\mu}{2\epsilon}\frac{x}{c}\right)$$

The right-going part of the eigensolution is

$$e^{-(\mu/2\epsilon)x/c}e^{s(t-x/c)}$$

which *decays exponentially* in the direction of propagation, and the left-going solution is

$$e^{(\mu/2\epsilon)x/c}e^{s(t+x/c)}$$

which also decays exponentially in the direction of travel.

Setting $s = j\omega$ and using superposition to build up arbitrary traveling wave shapes, we obtain the general class of solutions

$$y(t,x) = e^{-(\mu/2\epsilon)x/c} f\left(t - x/c\right) + e^{(\mu/2\epsilon)x/c} g\left(t + x/c\right)$$

Sampling these exponentially decaying traveling waves at intervals of T seconds (or $X = cT$ meters) gives

$$\begin{aligned}
y(t_n, x_m) &= e^{-(\mu/2\epsilon)x_m/c} f\left(t_n - x_m/c\right) + e^{(\mu/2\epsilon)x_m/c} g\left(t_n + x_m/c\right) \\
&= e^{-(\mu/2\epsilon)mX/c} f\left(nT - mX/c\right) + e^{(\mu/2\epsilon)mXssc} g\left(nT + mX/c\right) \\
&= e^{-\mu mT/2\epsilon} f\left[(n-m)T\right] + e^{\mu mT/2\epsilon} g\left[(n+m)T\right] \\
&= \left(e^{-\mu T/2\epsilon}\right)^m y^+(n-m) + \left(e^{\mu T/2\epsilon}\right)^m y^-(n+m)
\end{aligned}$$

The digital simulation diagram for this type of waveguide is shown in Figure 4.

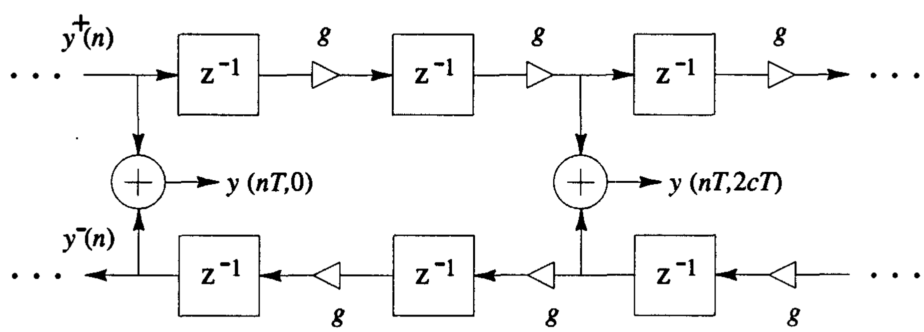

Figure 4. Discrete simulation of the ideal lossy waveguide. The loss factor $g = e^{-\mu T/2\epsilon}$ summarizes the distributed loss incurred in one sampling period.

Again note that the simulation is *exact* at the sampling points and instants, even though losses are admitted in the wave equation. Note also that the losses which are *distributed* in the continuous solution have been consolidated, or *lumped,* at discrete intervals of cT meters in the simulation. *The lumping of distributed losses does not introduce an approximation error at the sampling points.*

Consolidating Losses on a Larger Scale

In many applications, it is possible to realize vast computational savings in waveguide simulation by *commuting losses out of unobserved and undriven sections of the medium and consolidating them at a minimum number of points.* Because the digital simulation is linear and time invariant (given constant medium parameters K, ϵ, μ), and because linear, time-invariant elements commute, the diagram in Figure 5 is exactly equivalent (to within numerical precision) to the previous diagram in Fig. 4.

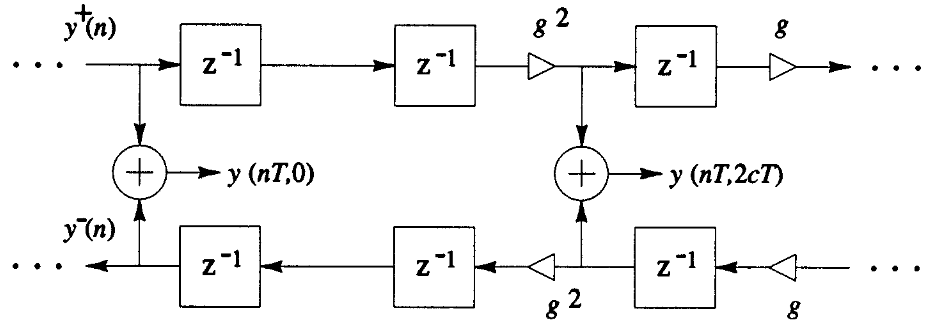

Figure 5. Discrete simulation of the ideal lossy waveguide. Each per-sample loss factor $g = e^{-\mu T/2\epsilon}$ is "pushed through" delay elements and combined with other loss factors until an input or output is encountered which inhibits further migration. If further consolidation is possible on the other side of a branching node, a loss factor can be pushed *through* the node by pushing a copy instance into each departing branch. If there are other inputs to the node, the *inverse* of the loss factor must appear on each of them. Similar remarks apply to pushing backwards through a node.

To illustrate how significant the computational savings can be, consider the simulation of a lossy "guitar string" for purposes of music synthesis. For simplicity, we rigidly terminate the length L string on either end which means traveling waves impinging on the "nut" or "bridge" will be reflected with a change of sign and no attenuation. The simulation diagram is given in Figure 6. Also, assume the string will be "plucked" from initial conditions so that there is no input to couple into the string. Let the output be simply the signal passing through a particular delay element rather than the more realistic summation of opposite elements in a bidirectional delay line.

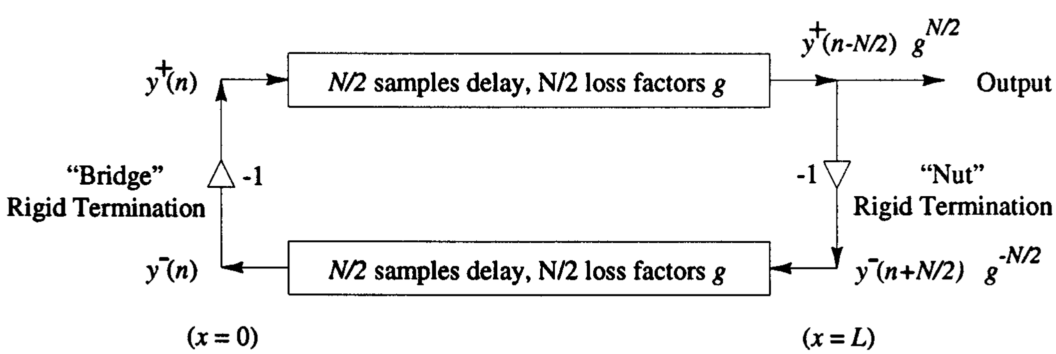

Figure 6. Discrete simulation of the rigidly terminated string with resistive losses. The N loss factors g are imbedded between the delay-line elements.

In this string simulator, there is a loop of delay containing $N = 2L/X = f_s/f_1$ samples where f_1 is the desired pitch of the string. Because there is no input/output coupling, we may lump *all* of the losses at a *single point* in the delay loop. Furthermore, the two reflecting terminations (gain factors of -1) may be commuted on top of each other to cancel them. Finally, the right-going delay may be combined with the left-going delay to give a single, length N, delay line. The result of these inaudible simplifications is shown in Figure 7.

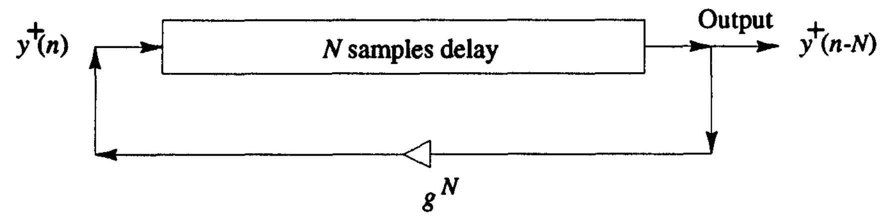

Figure 7. Discrete simulation of the rigidly terminated string with *consolidated* frequency-independent losses. All N loss factors g have been "pushed" through delay elements and combined at a *single* point.

If the sampling rate is $f_s = 50$ kHz and the desired pitch is $f_1 = 100$ Hz, the loop delay equals $N = 500$ samples. Since delay lines are efficiently implemented as circular buffers, (only a read-pointer and write-pointer need be used and updated each sample time per delay line), the cost of implementation is normally dominated by the loss factors, each one requiring a multiply every sample, in general.* Thus, *the consolidation of loss factors has reduced computational complexity by three orders of magnitude,* i.e., by a factor of 500 in this case. However, *the physical accuracy of the simulation has not been compromised.*

Frequency-Dependent Losses

It is well known that string losses increase with frequency. This is true whether the string is terminated rigidly or attached to a bridge. The largest losses in a real stringed instrument occur at the bridge. The bridge of a violin can be modeled up to about 5 kHz, for purposes of computing string loss, as a single spring in parallel with a frequency-independent resistance ("dashpot") (Cremer 1984, page 27). Similarly, air absorption increases with frequency, providing an increase in propagation loss for sound waves in air (Morse and Ingard 1968).

The derivation for the lossy waveguide can be repeated substituting $\mu(\omega)$ for μ, where ω is radian frequency in Hz. In other words, frequency-dependent losses can be treated exactly like frequency-independent losses. Mathematically, the frequency variable ω in the loss factor argument is treated as *independent* of the frequency variable $s \triangleq \sigma + j\omega$. Instead of factors g distributed throughout the diagram, the factors are now $G(\omega)$. These loss factors, implemented as digital filters, may also be consolidated at a minimum number of points in the waveguide without introducing an approximation error. However, an aproximation error will normally be introduced when replacing the irrational function $G(\omega)$ by a rational approximation which is realizable as a digital filter (Smith 1983).

In general, the coefficients of the optimal rational loss filter are obtained by minimizing $\| G(\omega) - \hat{G}(e^{j\omega}) \|$ with respect to the filter coefficients or the poles and zeros of the filter, where $\hat{G}(e^{j\omega})$ denotes the approximation filter frequency response. To avoid introducing frequency-dependent delay, the loss filter should be a *zero-phase, finite-impulse-response (FIR)* filter (Rabiner and Gold 1975).

Restriction to FIR filters yields the important advantage of keeping the approximation problem *convex* in the weighted least-squares norm. Convexity of a norm means that gradient-based search techniques can be used to find a global miminizer of the error norm without exhaustive search (Smith 1983).

Restriction to zero-phase loss filters means the impulse response $\hat{g}(n)$ must be symmetric about time zero, i.e., $\hat{g}(-n) = \hat{g}(n)$. (For real-time implementations, the filter can be made causal by "stealing" delay from the loop delay lines.) The zero-phase requirement halves the degrees of freedom in the filter coefficients. That is, given $\hat{g}(n)$ for $n \geq 0$, the coefficients $\hat{g}(-n) = \hat{g}(n)$ are also determined. The loss-filter frequency response can be written in terms of its (impulse response) coefficients as

$$\hat{G}(e^{j\omega}) = \hat{g}(0) + 2 \sum_{n=1}^{(N_{\hat{g}}-1)/2} \hat{g}(n) e^{j\omega n T}$$

* Losses of the form $1 - 2^{-k}$, $1 - 2^{-k} - 2^{-l}$, etc., can be efficiently implemented using shifts and adds.

where $\omega = 2\pi f_s = 2\pi/T$, and impulse-response length $N_{\hat{g}}$ is assumed odd.

A further degree of freedom is eliminated from the loss-filter approximation by assuming all losses are insignificant at 0 Hz so that $G(0) = 1 \Rightarrow \hat{G}(e^{j0}) = 1$ (taking the approximation error to be zero at $\omega = 0$). The condition $\hat{G}(1) = 0$ means the coefficients of the FIR filter must sum to 1. If the length of the filter is $N_{\hat{g}}$ (assumed odd), we have

$$1 = \sum_n \hat{g}(n) = \hat{g}(0) + 2 \sum_{n=1}^{(N_{\hat{g}}-1)/2} \hat{g}(n)$$

The *simplest* rational approximation $\hat{G}(z)$ is obtained with the *one-zero* digital filter.

$$\hat{G}(z) \triangleq z^{1/2} B(z) \triangleq b_0 z^{1/2} + b_1 z^{-1/2}$$

The zero-phase requirement gives rise to the factor $z^{1/2}$ and implies $b_0 = b_1$. (Since the length is even, time 0 must lie midway between the two impulse-response samples. In practice, we accept a half-sample delay from the one-zero filter and make up for it elsewhere in the loop.) Requiring no loss at $\omega = 0$ implies $b_0 + b_1 = 1$. Thus, two equations in two unknowns uniquely determine the coefficients to be $b_0 = b_1 = 1/2$ which gives a string loss frequency response equal to $\hat{G}(e^{j\omega}) = \cos\left(\frac{\omega T}{2}\right)$, $|\omega| \leq \pi f_s$. The simulation diagram for the ideal string with the simplest frequency-dependent loss filter is shown in Figure 8. Readers of the computer music literature will recognize this as the structure of the *Karplus-Strong algorithm* (Karplus and Strong 1983, Jaffe and Smith 1983).

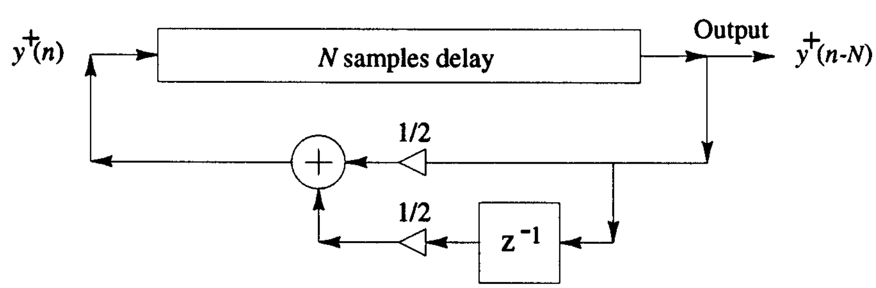

Figure 8. Rigidly terminated string with the simplest frequency-dependent loss filter. All N loss factors (possibly including losses due to yielding terminations) have been consolidated at a single point and replaced by a one-zero filter approximation.

The Karplus-Strong algorithm is obtained when the initial conditions used to "pluck" the string are obtained by filling the delay line with random numbers, or "white noise." Referring to Fig. 6, we know the initial *shape* of the string is obtained by *adding* the upper and lower delay lines, i.e., $y(t_n, x_m) = y^+(n - m) + y^-(n + m)$. It is straightforward to show that the initial *velocity* distribution along the string is determined by the *difference* between the upper and and lower delay lines. Thus, in the Karplus-Strong algorithm, the string is "plucked" by a completely *random initial displacement and initial velocity distribution.* This is a very energetic excitation, and usually in practice a lowpass filter is applied to the white noise to subdue it. We can see that by "drawing" an initial shape for the string twice in the loop delay, first in what was the upper delay-line and second in what was the lower delay-line in Fig. 6, we obtain a classical "plucked string" simulation with a zero initial velocity profile (Morse 1936). Similarly, by making the initial waveform in the lower delay-line the *negative* of the initial waveform in the upper delay-line, a classical "struck string" is obtained (Morse 1936).

The Dispersive One-Dimensional Wave Equation

Stiffness in a vibrating string introduces a restoring force proportional to the fourth derivative of the string displacement (Morse 1936):

$$\epsilon \ddot{y} = K y'' - \kappa y''''$$

where, for a cylindrical string of radius a and Young's modulus Q, the moment constant κ is equal to $\kappa = Q\pi a^4/4$. As before, we set $y(t,x) = e^{st+vx}$ to get

$$\epsilon s^2 = K v^2 - \kappa v^4$$

At very low frequencies, or when stiffness is negligible in comparison with K/v^2, we obtain again the non-stiff string: $\epsilon s^2 \approx K v^2 \;\Rightarrow\; v = \pm s/c$.

At very high frequencies, or when the tension K is negligible relative to κv^2, we obtain the *ideal bar* approximation

$$\epsilon s^2 \approx -\kappa v^4 \;\Rightarrow\; v \approx \pm e^{\pm j\frac{\pi}{4}} \left(\frac{\epsilon}{\kappa}\right)^{1/4} \sqrt{s}$$

The only restoring force in an ideal bar is due to stiffness. Setting $s = j\omega$ gives solutions $v = j\left(\epsilon/\kappa\right)^{1/4}\sqrt{\omega}$ and $v = \left(\epsilon/\kappa\right)^{1/4}\sqrt{\omega}$. In the first case, the wave velocity becomes proportional to $\sqrt{\omega}$. That is, waves travel faster along the ideal bar as oscillation frequency increases, going up as the square root of frequency. The second solution corresponds to a change in the wave shape which prevents sharp corners from forming due to stiffness (Cremer 1984).

At intermediate frequencies, between the ideal string and the ideal bar, the stiffness contribution can be treated as a correction term (Cremer 1984). This is the region of most practical interest because it is the principle operating region for strings, such as piano strings, whose stiffness has audible consequences (an inharmonic, stretched overtone series). We have, assuming $\kappa_0 \triangleq \kappa/K \ll 1$,

$$\begin{aligned}
s^2 &= \frac{K}{\epsilon}v^2 - \frac{\kappa}{\epsilon}v^4 = \frac{K}{\epsilon}v^2\left(1 - \frac{\kappa}{K}v^2\right) \triangleq c_0^2 v^2\left(1 - \kappa_0 v^2\right) \\
\Rightarrow\; v^2 &\approx \frac{s^2}{c_0^2}\left(1 + \kappa_0 v^2\right) \approx \frac{s^2}{c_0^2}\left(1 + \kappa_0 \frac{s^2}{c_0^2}\right) \\
\Rightarrow\; v &\approx \pm\frac{s}{c_0}\sqrt{1 + \kappa_0 \frac{s^2}{c_0^2}} \approx \pm\frac{s}{c_0}\left(1 + \frac{1}{2}\kappa_0 \frac{s^2}{c_0^2}\right)
\end{aligned}$$

Substituting for v in terms of s in s^{st+vx} gives the general eigensolution

$$e^{st+vx} = \exp\left\{ s\left[t \pm \frac{x}{c_0}\left(1 + \frac{1}{2}\kappa_0 \frac{s^2}{c_0^2}\right)\right]\right\}$$

Setting $s = j\omega$ as before, corresponding to driving the medium sinusoidally over time at frequency ω, the medium response is

$$e^{st+vx} = e^{j\omega[t \pm x/c(\omega)]}$$

where

$$c(\omega) \triangleq c_0\left(1 + \frac{\kappa\omega^2}{2Kc_0^2}\right)$$

Because the effective wave velocity depends on ω, we cannot use Fourier's theorem to construct arbitrary traveling shapes. At $x = 0$, we can construct any function of time, but the waveshape disperses as it propagates away from $x = 0$. The higher-frequency Fourier components travel faster than the lower-frequency components.

It is nevertheless possible to simulate the stiff medium in a local fashion. Since the temporal and spatial sampling intervals are related by $X = cT$, this must generalize to $X = c(\omega)T \Rightarrow T(\omega) = X/c(\omega) = c_0 T_0/c(\omega)$, where $T_0 = T(0)$ is the size of a unit delay in the absence of stiffness. Thus, a unit delay z^{-1} may be replaced by

$$z^{-1} \rightarrow z^{-c_0/c(\omega)} \qquad \text{(for frequency-dependent wave velocity)}$$

That is, each delay element becomes an *allpass filter* which approximates the required delay versus frequency. A diagram appears in Figure 9, where $H_a(z)$ denotes the allpass filter which provides a rational approximation to $z^{-c_0/c(\omega)}$.

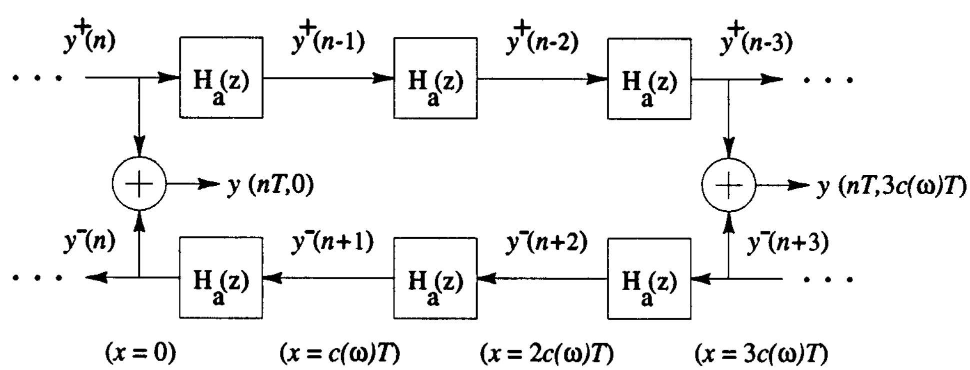

Figure 9. Section of a stiff string where the unit delay elements are replaced by allpass filters.

The general, order L, allpass filter is given by

$$H_a(z) \triangleq z^{-L} A(z^{-1})/A(z)$$

where

$$A(z) \triangleq 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_L z^{-L}$$

and the roots of $A(z)$ must all have modulus less than 1. That is, the numerator polynomial is just the reverse of the denominator polynomial.

For computability of the string simulation in the presence of scattering junctions, there must be at least one real sample of delay from the input to the output of at least one allpass filter along each uniform section of string. That is, for at least one allpass filter in each branch, we must have $H_a(\infty) = 0$ which implies $H_a(z)$ can be factored as $z^{-1}H_a'(z)$. In a systolic VLSI implementation, it is desirable to have at least one real delay from the input to the output of *every* allpass filter, in order to be able to pipeline the computation of all of the allpass filters in parallel. Computability can be arranged in practice by deciding on a minimum delay, (e.g., corresponding to the wave velocity at a maximum frequency), and using an allpass filter to provide excess delay beyond the minimum.

Because allpass filters are linear and time invariant, they commute like gain factors with other linear, time-invariant components. Figure 10 shows a diagram equivalent to Fig. 9 in which the allpass filters have been commuted and consolidated into two points. For computability, a single sample of delay is pulled out along each rail.

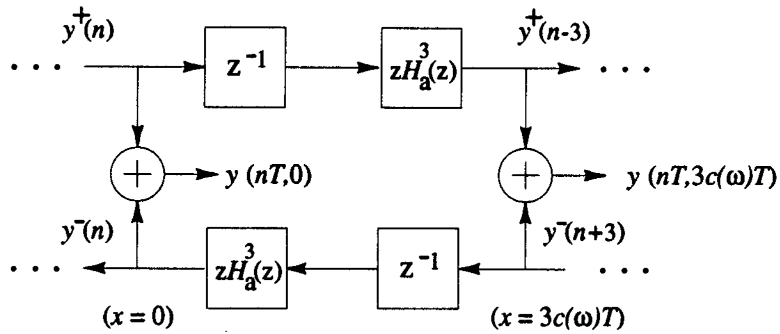

Figure 10. Section of a stiff string where the allpass delay elements are consolidated at two points, and a sample of pure delay is extracted from each allpass chain.

If $H_a(z)$ denotes the consolidated allpass transfer function, it can be designed by minimizing the *phase-delay error,* where phase delay is defined by

$$P(\omega) \triangleq -\frac{\angle H_a\left(e^{j\omega}\right)}{\omega} \qquad \text{(Phase Delay)}$$

It can be shown that minimizing the Chebyshev norm of the phase-delay error, $\| P(\omega) - c_0/c(\omega) \|_\infty$, approximates minimization of the error in *mode tuning* for the freely vibrating string (Smith 1983). Since the stretching of the overtone series is typically what we hear most in a stiff vibrating string, the worst-case phase-delay error is a good choice in musical acoustics.

A robust method for designing allpass filters with a prescribed group delay is given in (Yegnanarayana 1982). It is straightforward to adapt the method to approximate phase delay. To model stiff strings, the allpass filter must supply a phase delay which *decreases* as frequency increases. A good approximation may require a fairly high-order filter, adding significantly to the cost of simulation. It is therefore important to consolidate the loop delay as much as possible to minimize computational cost. However, note that to a large extent the allpass order required for a given error tolerance increases as the number of lumped frequency-dependent delays is increased.

Finally, for simple "piano string synthesis," the counterpart of Fig. 7 is shown in Figure 11 (Allen 1983). Of course, in practice, losses are also added for realistic behavior.

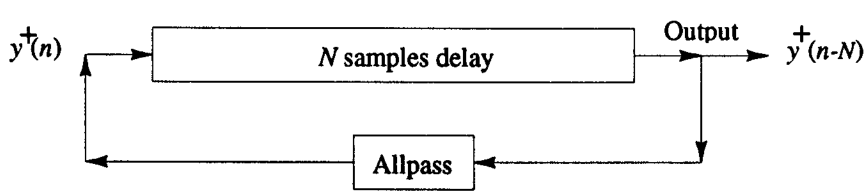

Figure 11. Simulation of a rigidly terminated, stiff string. All N allpass filters $H_a(z)$ have had a minimum sample of delay extracted, corresponding to the delay at $f_s/2$, and have been combined at a *single* point. The delay-line length N is the number of samples in K periods of f_K, where K is the number of the highest partial supported (normally the last one before $f_s/2$).

The General Linear Case

Given both frequency-dependent losses and dispersion, the wave equation is

$$\epsilon\ddot{y} + \mu(\omega)\dot{y} = Ky'' - \kappa y''''$$

In this case, the filter which replaces each delay element has a fully general frequency response

$$H_s(e^{j\omega}) \triangleq G(\omega)e^{-j\omega T c_0/c(\omega)}$$

Since every digital filter has a unique decomposition into a zero-phase filter in cascade with an allpass filter, we have $H_s(z)$ is simply a general digital filter transfer function. The general approximation problem is then to minimize $\| H_s(e^{j\omega}) - z^{m_s}\hat{H}_s(e^{j\omega}) \|$ with respect to the numerator and denominator coefficients of $\hat{H}_s(e^{j\omega})$, which are now largely unconstrained. (The filter must be stable, and we may wish to keep the condition $\hat{H}_s(1) = 1$. The term z^{m_s} corresponds to the delay which is "pulled out" of each generalized unit delay filter. Above, we suggested $m_s = c_0/c(\pi f_s)$ as a reasonable estimate for stiff strings. In order to allow the zero-phase part to span symmetrically to the left and right of the current time, a reasonable modification is to use $m_s = [(N_{\hat{g}} - 1)/2][c_0/c(\omega_0)]$ which shifts the impulse response left by half the natural delay of the zero-phase part at some intermediate frequency ω_0.

Since general, linear, time-invariant filters commute, the simulation diagram can be simplified exactly as in figures 10 and 11. A qualitative depiction of the general case is shown in Figure 12.

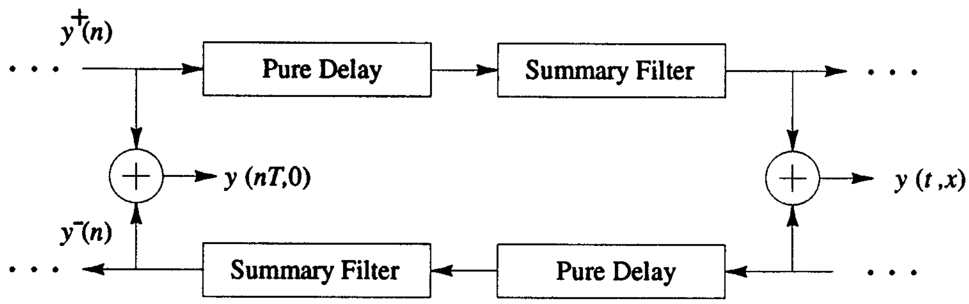

Figure 12. Simulation of a general string section. All distributed losses and dispersion are "swept out" of the string, leaving pure delay behind, and are consolidated, or "lumped," into discrete "summary filters."

Concerning higher order terms in the wave equation, the general, linear, time-invariant, differential equation is

$$\sum_{k=0}^{\infty} \alpha_k \frac{\partial^k y(t,x)}{\partial t^k} = \sum_{l=0}^{\infty} \beta_l \frac{\partial^l y(t,x)}{\partial x^l}$$

which yields the algebraic equation,

$$\sum_{k=0}^{\infty} \alpha_k s^k = \sum_{l=0}^{\infty} \beta_l v^l$$

Solving for v in terms of s is, of course, nontrivial. However, in specific cases, it is plausible to expect one could determine the appropriate $G(\omega)$ and $c(\omega)$ by numerical map construction. For example, starting at $s = 0$, we have also $v = 0$. Stepping s forward by a small differential $j\Delta\omega$, the left-hand side can be approximated by $\alpha_0 + \alpha_1\Delta\omega$. Requiring the generalized wave velocity $s/v(s)$ to be continuous, a physically reasonable assumption, the right-hand side can be approximated by $\beta_0 + \beta_1\Delta v$, and the solution is easy. As s steps forward, higher order terms become important one by one on both sides of the equation. Each new term in v spawns a new solution for v in terms of s, since the order of the polynomial in v is incremented. It is quite possible that *homotopy continuation methods* (Morgan 1987) can be used to keep track of the branching solutions of v as a function of s. For each solution $v(s)$, let $v_r(\omega)$ denote the real part of $v(j\omega)$ and let $v_i(\omega)$ denote the imaginary part. Then the solution family can be seen in the form $\exp\{j\omega t + v(j\omega)x\} = \exp\{\pm v_r(\omega)x\} \cdot \exp\{j\omega\,(t \pm v_i(\omega)x/\omega)\}$. Sampling gives $\exp\{\pm v_r(\omega)x_m\} \cdot \exp\{j\omega\,(t_n \pm x_m/c(\omega))\} = [\exp\{\pm v_r(\omega)X\}]^m \cdot \exp\{j\omega\,(n \pm m)\,T(\omega)\}$, where $c(\omega) \triangleq \omega/v_i(\omega)$, and $X = c(\omega)T(\omega)$ as before. Thus, a completely general map of v versus s, corresponding to a partial differential equation of any order, can be translated, in principle, into an accurate, local, linear, time-invariant, discrete-time simulation.

Linear Extensions

Beyond the vibrating medium itself, there are couplings to the outside world to consider. For example, a violin model would require attachment of the string to a digital filter which represents the body of the violin (Cremer 1984, Smith 1983). The violin body, would then radiate into a room model (reverberator), etc. In a piano simulation, it is desirable to couple strings together (Weinreich 1979, Jaffe and Smith 1983). Even to simulate a *single* string properly requires simulation of two transverse modes of vibration, in the vertical and horizontal planes; these modes couple together linearly through the bridge, and nonlinearly along the string via neglected terms in the differential equations (Elmore and Heald 1969). In the bowed string, *torsional waves* are known to be important (Cremer 1984, McIntyre, Schumacher, and Woodhouse 1983).

In practical simulations of musical strings and air columns, there is need of another allpass filter in the delay loop which provides *fine tuning* of the perceived pitch of the vibration. A tuning technique using a first-order allpass filter is described in (Jaffe and Smith 1983). Also in that paper are techniques for simulating a moving "pick," sympathetically vibrating strings, providing a "dynamic level" illusion, and other items of interest to the musician. The tuning-allpass technique can be extended to provide smooth *vibrato* as though the player were moving a finger along the fingerboard (Smith 1983).

Nonlinear Extensions

A first set of nonlinear extensions has to do with nonlinear coupling such as occurs in the *bowed string* and *woodwinds*. The basic theory of the bowed string is covered in (Cremer 1984), and in a unified fashion for bowed strings, single-reed woodwinds, and jet-driven instruments, such as the flute and organ pipe, by (McIntyre, Schumacher, and Woodhouse 1983).

In this paper, only linear systems arising from a linear set of differential equations have been discussed. However, the very fact that the simulation produces physically meaningful variables makes it straightforward to extend to nonlinear behavior. The general technique is to "watch" some variable, such as displacement of the string, or instantaneous pressure in the air column, and modify the otherwise constant coefficients in the differential equation for the medium. Below is a list of some of the possible applications of this approach:

- Simple "soft clipping" can be implemented for both strings and air columns by watching the displacement and attenuating it when it reaches the elastic or adiabatic limit of the medium. In other words, the loss factors can be made amplitude-dependent.
- String tension increase causes pitch increase. The delay in a section of string can be reduced as displacement is seen to increase, or the integral of the displacement can be used to control a more global sampling rate. In strings with low tension, the initial decay of the pitch gives a "twang" to the sound.
- The bow-string theory does not provide a restoring force due to tension increase caused by string elongation. By watching the string length as is done for pitch-sharpening simulation, the proper restoring force can be added. Without this force, it is easy for the string to "stick" to the bow and depart with it forever.
- In stiff string simulation, the "curvature" can be watched, and when it increases beyond a certain point, a corrective exponential shape can be added to the current string displacement, thus "rounding" what is becoming a "corner" in the string. This technique compensates for having neglected the exponentially decaying solutions to the differential equation for the stiff string.
- In air columns, local increases in pressure cause faster propagation of sound. (In strings, on the other hand, local tension increases are so rapidly propagated throughout the length of the string that they are best regarded as global tension increases.) By shrinking the unit delays in regions of the simulation where instantaneous pressure is high, according to standard first-order approximations for nonlinear propagation in air (Beyer 1974), the sharpening of wave crests as in "shock wave" formation can be simulated. Such an effect may be important in the simulation of brasses and other instruments where the pressure near the mouthpiece gets large enough to enter the nonlinear regime.

- Bending hysteresis in strings can be simulated by watching the transverse displacement and velocity and applying a force correction.
- Nonlinear mode coupling of string vibration modes can be implemented.
- From the absolute flow and known area function in an air column or "acoustic tube," it is possible to predict the formation of turbulence at a constriction. This might enable higher quality simulation of voice, flutes, and the like, by automatically injecting the proper filtered noise at the right place in the simulation.
- From the absolute flow in an air column, the pressure due to the Bernoulli effect can be computed. This pressure can be used to compute forces on the walls with practical applications such as pulling reeds together in a bassoon model or the vocal folds together in a vocal synthesis simulation (Cook 1990).
- From the absolute flow and detailed acoustic geometry, the *type* of flow can be predicted, i.e., whether it is laminar, turbulent, vortex shedding, etc., and where jets may form, where flow separates from a wall, and so on. Some relevant observations appear in (Hirschberg 1990). Knowledge of flow details can provide the information needed to add losses, delay modulation, radiators, and/or filtered noise in order to maintain numerous acoustically important phenomena.

References

J. B. Allen, private communication.

Nonlinear Acoustics, Naval Ship Systems Command, Dept. of the Navy, 1974.

"SPASM: A Real-Time Vocal Tract Physical Model Editor, Controller, and Singer: The Companion Software Synthesis System," Colloquium on Physical Modeling, Grenoble, September 1990.

L. Cremer, *The Physics of the Violin,* MIT Press, Cambridge MA, 1984.

W. C. Elmore and M. A. Heald, *Physics of Waves,* Dover, New York, 1969.

L. Hiller and P. Ruiz, "Synthesizing Musical Sounds by Solving the Wave Equation for Vibrating Objects," *J. Audio Eng. Soc.,* Part I: vol. 19, no. 6, June 1971; Part II: vol. 19, no. 7, July/Aug. 1971.

A. Hirschberg, R. W. A. van de Laar, J. P. Marrou-Maurieres, A. P. J. Wijnands, H. J. Danes, S. G. Kruijswijk, and A. J. M. Houtsma, "A Quasi-Stationary Model of Air Flow in the Reed Channel of Single-Reed Woodwind Instruments," *Acustica*, vol. 70, pp. 146--154, 1990.

D. Jaffe and J. O. Smith, "Extensions of the Karplus-Strong Plucked String Algorithm," *Computer Music J.*, vol. 7, no. 2, pp. 56--69, 1983.

K. Karplus and A. Strong, "Digital Synthesis of Plucked String and Drum Timbres," *Computer Music J.*, vol. 7, no. 2, pp. --55, 1983.

D. H. Keefe, "On Sound Production in Reed-Driven Wind Instruments," Submitted to the Journal of the Acoustical Society of America.

N. J. Loy, *An Engineer's Guide to FIR Digital Filters,* Prentice-Hall Inc., Englewood Cliffs, NJ, 1988.

M. E. McIntyre, R. T. Schumacher, and J. Woodhouse, "On the Oscillations of Musical Instruments," *J. Acoust. Soc. Amer.*, vol. 74, no. 5, pp. 1325--1345, Nov. 1983.

A. Morgan, *Solving Polynomial Systems Using Continuation for Engineering and Scientific Computations,* Prentice-Hall Inc., Englewood Cliffs, NJ, 1987.

P. M. Morse, *Vibration and Sound,* published by the American Institute of Physics for the Acoustical Society of America, 1976 (1st ed. 1936, 2nd ed. 1948).

P. M. Morse and K. U. Ingard, *Theoretical Acoustics,* McGraw-Hill, New York, 1968.

L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing,* Prentice-Hall Inc., Englewood Cliffs, NJ, 1975.

P. M. Ruiz, "A Technique for Simulating the Vibrations of Strings with a Digital Computer," *M. Music Diss.,* Univ. Ill., Urbana, 1969.

J. O. Smith, "Techniques for Digital Filter Design and System Identification with Application to the Violin," Ph.D. Dissertation, Elec. Eng. Dept., Stanford University, June 1983.

J. O. Smith and P. Gossett, "A Flexible Sampling-Rate Conversion Method," *Proc. IEEE Conf. Acoust. Sp. and Sig. Proc., vol. 2, pp. 19.4.1-19.4.2, San Diego, March 1984.*

J. O. Smith, "A New Approach to Digital Reverberation using Closed Waveguide Networks," *Proc. 1985 Int. Conf. Computer Music,* Vancouver Canada, Computer Music Assoc., 1985. Music Dept. Tech. Rep. STAN--M--31, Stanford University, July 1985.

J. O. Smith, "Elimination of Limit Cycles and Overflow Oscillations in Time-Varying Lattice and Ladder Digital Filters," Music Dept. Tech. Rep. STAN--M--35, Stanford University, May 1986. Shorter version, "Elimination of Limit Cycles in Time-Varying Lattice/Ladder Filters," in *Proc. IEEE Conf. Circuits and Systems*, San Jose, May 1986.

J. O. Smith, "Efficient Simulation of the Reed-Bore and Bow-String Mechanisms," *Proc. 1986 Int. Conf. Computer Music,*, The Hague, Netherlands.

J. Wawrzynek, et al., "A VLSI Approach to Sound Sythesis," *Proc. 1984 Int. Computer Music Conf.,* Computer Music Assoc., San Francisco, 1984.

G. Weinreich, "Coupled Piano Strings," *J. Acoust. Soc. Amer.*, vol. 62, no. 6, pp. 1474--1484, Dec. 1977. Also see *Scientific American,* vol. 240, p. 94, 1979.

B. Yegnanarayana, "Design of Recursive Group-Delay Filters by Autoregressive Modeling," *IEEE Trans. on Acoust., Speech, and Signal Proc.*, vol. ASSP--30, pp. 632--637, Aug. 1982.