Chapter 3

Sound modeling: source-based approaches

Federico Avanzini

3.1 Introduction

It was 1971 when Hiller and Ruiz envisioned the possibility of using numerical simulations of the wave equation for sound synthesis applications.

> [...] This is a completely new approach to electronic sound synthesis insofar as the starting point is the physical description of the vibrating object [...]

A decade later McIntyre, Schumacher, and Woodhouse published their classic study on the use of non-linear maps for modeling the generation of self-sustained oscillations in musical instruments.

> [...] a fast minicomputer could produce results at a cycle rate in the audible range. The result would perhaps have some novelty: an electronic musical instrument based on a mathematical model of an acoustic instrument [...]

Today the algorithms described by these authors can be easily implemented in real-time on general-purpose hardware, and it is common practice to use the term *physical modeling* to refer to sound modeling techniques in which the synthesis algorithms are designed based on a description of the physical phenomena involved in sound generation.

Direct sound representations, that are merely based on a description of the sound waveform, do not contain information about the way the sound has been generated and processed by the surrounding environment before arriving to the listener's ear. Sampling in time the sound signal does not assume any underlying structure, or process, or generative model, in sound representation. The symbolic description is extremely poor, and as a consequence very little interaction with the sound representations is allowed. Although signal processing techniques can provide meaningful modifications (e.g. pitch shift, time stretching), sampling is basically a *static*, low-level description of sound.

High level representations of sound signals are necessarily associated with some abstract paradigms that underlie sound production. As we have seen previously, when trying to develop a taxonomy of sound synthesis methods a first distinction can be traced between *signal models* and *source models*. Any algorithm which is based on a description of the sound pressure signal and makes no assumptions on the generation mechanisms belongs to the class of signal models. Additive synthesis is a good example of a signal model: as already mentioned, one major drawback of this technique is its enormous number of control parameters: at least one amplitude and one pitch envelopes have to be specified for each partial. Moreover, the sound representation has not a strong *semantic* interpretation, since these parameters do not have a high-level meaning. Subtractive synthesis with its source-filter structure provides in a sense a more semantic description of sound: in certain cases the two blocks can be given a physical interpretation in terms of an exciting action and a resonating object, respectively. As an example, in the case of LPC based speech synthesis the broadband input signal can be interpreted as a glottal source signal, and the shaping filter represents the action of the vocal tract. However, in many other cases this interpretation does not hold, and the control parameters in the model (e.g., the filter coefficients) do not have a high-level meaning.

Source models aim at describing the physical objects and interactions that have generated an acoustic event rather than the acoustic signal itself. This modeling approach often gives rise to rather complex descriptions, that can lead to computationally expensive numerical algorithms. Several modeling paradigms and techniques are available in the literature for deriving efficient implementations of such descriptions, including lumped/distributed modeling, waveguide structures, finite difference methods, and so on. The following sections describe in detail a few of these approaches. Here it is worth discussing another aspect, i.e. that of control. A direct consequence of assuming a source-based approach is that the resulting control parameters have a straightforward physical interpretation: typical parameters in the models are associated with masses, hardness/softness characteristics, blowing pressures, lengths: such a semantic representation can in principle allow more intuitive interaction.

Source-based sound modeling paradigms are often grouped into two broad categories, namely *lumped* and *distributed* models. Generally speaking, distributed models are more often used for describing vibrating bodies or air volumes where forces and matter depend on (and propagate along) both time and space. One-, two- and three-dimensional resonators (such as strings, bars, acoustical bores, membranes, plates, rooms, etc.) can be treated as continuous distributed systems, and mathematically described by means of Partial Differential Equations (*PDE*s). One of the most popular distributed modeling approaches is *waveguide modeling*, which will be discussed in detailed in Sec. 3.4.

Although waveguides are extremely successful in modeling nearly elastic mediums, where the D'Alembert equation or some of its generalizations hold, they are not equally good in dealing with systems where these hypotesis are not met. As an example, oscillations in a bar are governed by the so called Euler-Bernoulli equation, for which no traveling-waves schematization can be assumed. One possible approach for dealing with such systems is using *finite difference* or *finite elements* methods. These time-domain techniques are based on direct discretization of the PDEs and consequently have high computational costs. On the other hand, when properly applied they provide stable and very accurate numerical systems.

As opposed to distributed models, lumped models are used when a physical system can be conveniently described without explicitly considering its extension in space. As an example, a mechanical resonating body may be described in terms of ideal masses or rigid elements, connected to each other with spring and dampers, and possibly non-linear elements. Similar considerations may apply to electrical circuits and even to certain acoustic systems. The resulting models are naturally described in the time domain, in terms of Ordinary Differential Equations (*ODE*s). Sec. 3.5 discusses lumped modeling approaches, and includes an introduction to modal synthesis. Defining modal synthesis as a lumped modeling approach may be questionable, since the modal formalism incorporates a "spatial" representation

(e.g. it is possible to inject a force in a specific point of a modal resonator, or to measure its displacement in a specific point). On the other hand, representing a resonator as a combination of a finite number of modes correspons to approximating the resonator as a mesh of point masses connected with strings and dampers, and in this sense modal synthesis may be regarded as a lumped modeling approach.

3.2 Physical structures and models

3.2.1 Simple vibrating systems and normal modes

Sound is produced by mechanical, acoustical, or electrical vibrations that ultimately generate an acoustic pressure signal that reaches our ear. In this section we review the most elementary oscillating systems and their properties.

3.2.1.1 Oscillators

The simplest physical oscillating system is the damped second-order (or harmonic) oscillator. A generic oscillator of this kind is described by the following linear differential equation:

$$\ddot{x} + 2\alpha\dot{x} + \omega_0^2 x = u_{\text{ext}}(t), \tag{3.1}$$

where u_{ext} is an external driving signal. The general solution of the homogeneous equation (i.e., Eq. (3.1) with $u_{\text{ext}} = 0$) is given by

$$x(t) = a_0 e^{-\alpha t}\cos(\omega_r t + \phi_0), \tag{3.2}$$

where $\omega_r = \sqrt{\omega_0^2 - \alpha^2}$. The parameters a_0 and ϕ_0 are uniquely determined by the initial conditions $x(0),\ \dot{x}(0)$. In particular the impulse response of the system corresponds to initial conditions $x(0) = 0$ and $\dot{x}(0) = 1$, and is given by $h(t) = e^{-\alpha t}\sin(\omega_r t)/\omega_r$.

An electrical system representing a damped harmonic oscillator is the RLC circuit (Fig. 3.1(a)).

$$L\frac{d^2 i}{dt^2}(t) + R\frac{di}{dt} + \frac{1}{C}i = 0, \tag{3.3}$$

where i is the current in the circuit, $L,\ R,\ C$ are the inductance, resistance, and capacitance of in the circuit, respectively. This is an equation of the form (3.1), with $\alpha = R/2L$, $\omega_0^2 = 1/LC$. Therefore it has a solution of the form (3.2).

In the mechanical case, an instance of damped harmonic oscillator is the mass-spring-damper system depicted in Fig. 3.1(b):

$$m\ddot{x}(t) + r\dot{x}(t) + kx(t) = 0, \tag{3.4}$$

where x is the displacement signal, $m,\ r,\ k$ are the mass, mechanical resistance, and spring stiffness. Again this is an equation of the form (3.1), with $\alpha = r/2m$, $\omega_0^2 = k/m$. Therefore it has a solution of the form (3.2).

In certain situations, acoustic systems can also be described in terms of lumped elements that are equivalent to resistance, capacitance, and inductance. The variables involved in this case are air-flow (or volume velocity) $u(t)$, measured in m^3/s, and acoustic pressure $p(t)$, measured in Pa. When the dimensions of an acoustic element are much less than the sound wavelength, then the acoustic pressure, p can be assumed constant over the element. In this case, the acoustic behavior of the element is, at least at low frequencies, very simple. In particular, the Helmholtz resonator (depicted in Fig. 3.1(c)) behaves to a good degree of approximation as a second order oscillator. We analyze this systems in terms of three main elements: the opening, the neck, and the cavity.

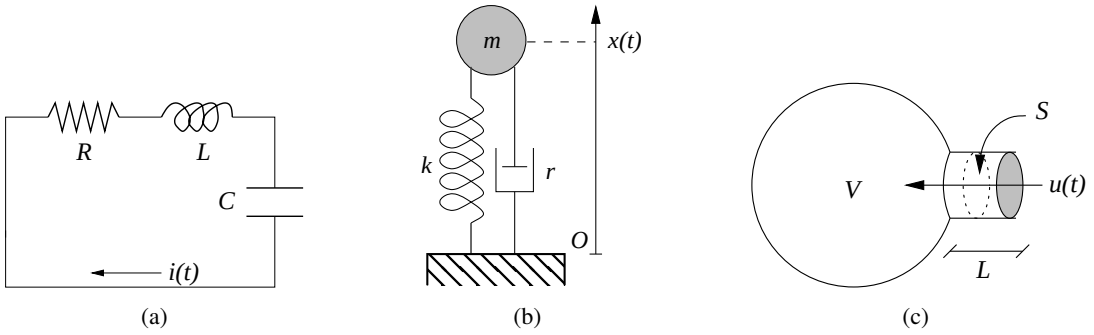


Figure 3.1: *Second order electrical, mechanical, and acoustic oscillators; (a) a RLC circuit; (b) a mass-spring-damper system; (c) a Helmholtz resonator.*

Resistive phenomena are observed during the passage of acoustic airflow through the opening, due to a pressure difference $\Delta p_{\text{op}}(t)$: the flow behavior is dominated by viscous and thermal losses and it is reasonably assumed to be in phase with the acoustic pressure. Therefore the relation $\Delta p_{\text{op}}(t) = Ru(t)$ holds at the opening where the constant R is termed *fluid-dynamic resistance*. Simple *inertial* behaviors are observed in the cylindrical neck. The air mass inside this tube is $m = \rho_{air} SL$ (ρ_{air} being the air density, S the cross-sectional area, and L the length). If a pressure difference $\Delta p_{\text{tube}}(t)$ is applied at the tube ends, the enclosed air behaves like a lumped mass driven by the force Δp_{tube}, and Newton's law implies

$$\Delta p_{\text{tube}}(t) = \rho_{air} SL \cdot \dot{v}(t), \tag{3.5}$$

where the relation $u(t) = Sv(t)$ has been used, and $v(t)$ indicates particle velocity. Finally, the cavity has an elastic behavior. Consider the volume $V(t)$ of air inside the cavity: the contraction $dV(t)$ caused by a pressure difference $\Delta p_{\text{cav}}(t)$ is such that $-\rho_{air} c^2 \cdot dV/V = \Delta p_{\text{cav}}$. As a consequence, a new air volume $-dV$ can enter the cavity. By definition, this equals the integral of $u(t)$ over time, therefore

$$\begin{aligned} -dV(t) &= \int_0^t u(t')dt' = \frac{V}{\rho_{air} c^2} \Delta p_{\text{cav}}(t). \\ S\Delta p_{\text{cav}}(t) &= -\frac{\rho_{air} S^2 c^2}{V} \int_0^t v(t')dt', \end{aligned} \tag{3.6}$$

which represent a linear spring with stiffness $\rho_{air} S^2 c^2 / V$. Both the air mass in the tube and the resistance at the opening impede the same flow u, and are therefore in a "series" connection. This flow u enters the cavity, so that the the volume is in series with the other two. The resulting equation for the particle displacement x is

$$(\rho_{air} SL) \cdot \ddot{x}(t) + R\dot{x}(t) + \frac{\rho_{air} S^2 c^2}{V} x(t) = 0. \tag{3.7}$$

Again equation of the form (3.1). Therefore solution of the form (3.2).

3.2.1.2 Impedance

The examples in the previous section show that in a large class of systems it is possible to construct pairs of variables (often defined as *Kirchoff variables*) with the property that their product has the dimensions of power (Kg m^2/s^3). In electrical systems such a pair of variables is given by (v, i), voltage and current. Integro-differential relations can be found that relate these two variables, in particular three elementary

Electrical		Mechanical		Acoustic	
Current i (A)		Velocity v (m/s)		Flow u (m^3/s)	
Voltage v (V)		Force f (N)		Pressure p (Pa)	
(Resistance) R	$\left(\frac{\text{Kg}\cdot\text{m}^2}{\text{s}}\right)$	(Damping) r	$\left(\frac{\text{Kg}}{\text{s}}\right)$	(Opening) R	$\left(\frac{\text{Kg}}{\text{m}^4\cdot\text{s}}\right)$
(Capacitance) $1/sC$		(Spring) k/s		(Cavity) $\rho_{air}c^2/Vs$	
(Inductance) s/L		(Mass) $m \cdot s$		(Bore) $\rho_{air}Ls/S$	

Table 3.1: *Summary of analogies in electrical, mechanical and acoustical systems.*

relations define the fundamental quantities resistance R, inductance L and capacitance C. In the Laplace domain, the integro-differential equations are turned into simple algebraic relations:

$$V(s) = R \cdot I(s), \qquad V(s) = sL \cdot I(s), \qquad V(s) = \frac{1}{sC} I(s). \tag{3.8}$$

These are particular examples of a more general relation in linear electric circuits:

$$V(s) = Z(s)I(s), \tag{3.9}$$

where the quantity $Z(s)$ is called *impedance* of the circuit and is defined as the ratio between the Laplace transforms of voltage and current intensity. The inverse of $Z(s)$ is called *admittance*, and it is usually denoted as $\Gamma(s) = Z(s)^{-1}$.

Similar considerations apply to mechanical systems. Force f (Kg m/s^2) and velocity v (m/s) are the mechanical Kirchhoff variables, since their product is a power. Again, the ratio of these two variables in the Laplace domain is defined as (mechanical) *impedance*, and its inverse is the (mechanical) admittance. In the mechanical oscillator described above we have already introduced the three mechanical equivalents of resistance, capacitance and inductance. The direct proportionality $f(t) = rv(t)$ defines ideal linear viscous forces, and by comparison with the first of Eqs. (3.8) r can be regarded as a mechanical resistance. The inertial mass m of a non-relativistic body is defined as the ratio between the total force acting on it and its acceleration, i.e. $f(t) = ma(t) = m\dot{v}(t)$, or $F(s) = msV(s)$ in the Laplace domain, and by comparison with the second equation in (3.8) m can be regarded as a mechanical inductance. Finally, in an ideal linear spring the elastic force is proportional to the elongation of the spring: $f(t) = kx(t) = k\int_0^t v(t')dt'$. or $F(s) = k/s\ V(s)$ in the Laplace domain, and by comparison with the third equation in (3.8) the stiffness k can be regarded as a mechanical capacitance. Therefore the aggregate impedance $Z(s)$ of a second-order mechanical oscillator is $Z(s) = ms + k/s + r$.

As far as acoustic systems are concerned, acoustic pressure p (Kg/ms^2) and volume velocity u (m^3/s) are the acoustic Kirchhoff variables, since their product is a power. Again, the ratio of these two variables in the Laplace domain is defined as (acoustic) *impedance*, and its inverse is the (acoustic) admittance. In the Helmoltz resonator described above we have already introduced the three acoustic equivalents of resistance, capacitance and inductance. More precisely, fluid-dynamic resistance is associated to viscous and thermal losses at narrow openings: $p(t) = Ru(t)$. Fluid-dynamic *inductance* is associated to short, open tubes: $p(t) = \rho_{air}L/S \cdot \dot{u}(t)$, or $P(s) = \rho_{air}Ls/S \cdot U(s)$ in the Laplace domain. Fluid-dynamic *capacitance* is associated with enclosed air volumes: $p(t) = \rho_{air}c^2/V \cdot \int u(t')dt'$, or $P(s) = \rho_{air}c^2/(Vs) \cdot U(s)$ in the Laplace domain.

Table 3.1 summarizes the main analogies between electrical, mechanical, and acoustic systems, that we have discussed throughout this section.

3.2.1.3 Coupled oscillators and modal decomposition

We have examined above the behavior of a single second-order oscillator. A way of describing more complex oscillating systems is to represent them as combinations of the simple elements described so far. As an example an oscillating mechanical system (a string, a membrane, etc.) can be described in terms of point masses coupled through linear springs and dampers. Therefore the system is described as a set of coupled second-order differential equations.

It is known that, under general hypotheses, one can find a change of variables such that the set of coupled equations is turned into a set of *uncoupled* second-order equations. In order to clarify this concept, let us look at the following simple mechanical example in which two point masses are connected to each other and to the "walls" through three springs:

$$\begin{aligned} m\ddot{x}_1(t) + kx_1(t) + k(x_1 - x_2) &= 0, \\ m\ddot{x}_2(t) + kx_2(t) + k(x_2 - x_1) &= 0. \end{aligned} \tag{3.10}$$

If we introduce a suitable set of variables $q_{1,2}$ in place of $x_{1,2}$, the above equations can be decoupled, or diagonalized:

$$\begin{aligned} \ddot{q}_1(t) &= -\omega_0^2 q_1(t), \\ \ddot{q}_2(t) &= -3\omega_0^2 q_2(t), \end{aligned} \tag{3.11}$$

with $q_1 = x_1 + x_2$, $q_2 = x_1 - x_2$, $\omega_0^2 = k/m$. The *normal modes* q_i ($i = 1, 2$) are uncoupled and the x_i are linear combinations of the q_i.

This simple example can be extended to more complicated systems, composed N masses coupled through springs and dampers. One can in general reformulate the system in terms of *normal modes* of oscillation, and the oscillation of each point mass can be seen as a linear combination of N normal modes, each of which obeys the equation of a second-order (damped) harmonic oscillator. We will return on these concept in Sec. 3.5.2, when discussing modal synthesis.

3.2.2 Continuous vibrating systems and waves

In the previous section we have examined oscillating systems constructed with lumped elements (e.g. resistances, capacitances, inductances, and their mechanical and acoustic counterparts), and are therefore represent by a finite and discrete set of points in space (e.g. a set of point masses). In this section we examine vibrating systems that are distributed continuously in space, and are therefore described by partial differential equations involving both space and time, rather than set of ordinary differential equations in time.

3.2.2.1 The one-dimensional D'Alembert equation

Vibrational phenomena in an ideal *elastic* medium are described by the D'Alembert equation, whose one-dimensional version is written as

$$\frac{\partial^2 y}{\partial x^2}(x,t) = \frac{1}{c^2}\frac{\partial^2 y}{\partial t^2}(x,t). \tag{3.12}$$

This equation holds, for instance, in an ideal string of length L, linear mass density μ and tension T. In this case the variable $x \in [0, L]$ stands for position along string length and y stands for *transversal* displacement of the string. The constant c has the value $\sqrt{T/\mu}$ and has the dimensions m/s of a velocity. A full derivation of Eq. (3.12) for the ideal string can be found in many textbooks: roughly speaking, the

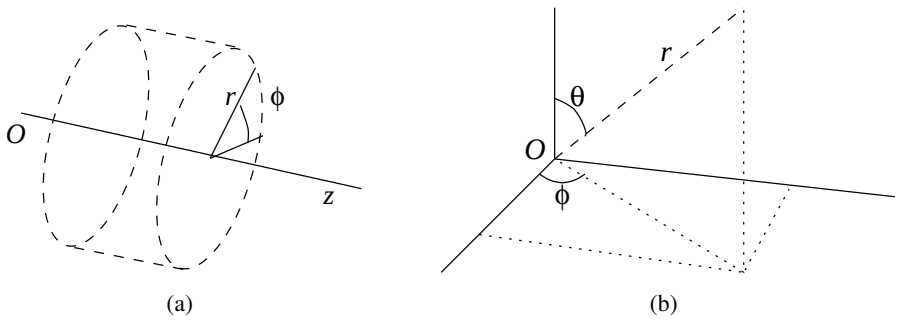


Figure 3.2: *Illustration of (a) cylindrical and (b) spherical coordinates.*

two main assumptions are that (i) the infinitesimal string segment dx moves only in the vertical direction, so that its acceleration can be computed using only the transverse component of the tension as the acting force; and (ii) the amplitude of the vibrations is very small.

There are interesting cases where acoustic disturbances can be assumed to be one-dimensional up to a reasonable approximation. Propagation of acoustic pressure in a cylindrical or in a conical tube is an example. Using cylindrical coordinates (see Fig. 3.2(a)), one can show that for cylindrical bores one-dimensional *longitudinal* pressure waves in the z direction are described using Eq. (3.12), with z in place of x and with y representing acoustic pressure. Using spherical coordinates (see Fig. 3.2(b)), one can show that for conical bores one-dimensional *spherical* pressure waves are described through the equation

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial R}{\partial r}\right)(r,t) = \frac{1}{c^2}\frac{\partial^2 R}{\partial t^2}(r,t), \tag{3.13}$$

in which $R(r)$ represents acoustic pressure, and the Laplacian operator is expressed in spherical coordinates as $\nabla^2 = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial}{\partial r}\right) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial}{\partial\theta}\right) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2}{\partial\phi^2}$. Using the substitution $R = \tilde{R}/r$, it is easily seen that Eq. (3.13) reduces to the one dimensional D'Alembert equation (3.12) for the variable $\tilde{R}$.

3.2.2.2 Traveling wave solution

A fundamental property of Eq. (3.12) is that it describes *propagation* phenomena. This statement can by proved by factoring the equation as follows:

$$\left(\frac{\partial}{\partial x} - \frac{1}{c}\frac{\partial}{\partial t}\right)\left(\frac{\partial}{\partial x} + \frac{1}{c}\frac{\partial}{\partial t}\right)y = 0. \tag{3.14}$$

From this factorization it is easily seen that generic solutions take the form

$$y(x,t) = y^+(ct - x) + y^-(ct + x). \tag{3.15}$$

This is the solution to Eq. (3.12) originally proposed by D'Alembert himself. The two functions $y^\pm$ describe waveforms that translate rigidly with velocity c, in the right-going and left-going directions, respectively. Their shape is determined by the boundary conditions (in space) and the initial conditions (in time). As an example, if y represents the displacement of a vibrating string the initial conditions are represented by an initial displacement and an initial velocity:

$$y_0(x) = y(x,0), \qquad v_0(x) = \frac{\partial y}{\partial t}(x,0). \tag{3.16}$$

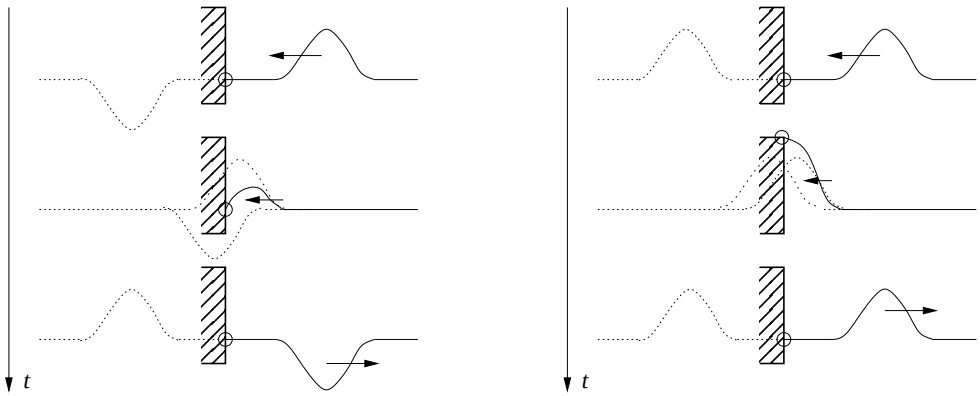


Figure 3.3: *Boundary conditions and wave reflections; (a) fixed string end and negative wave reflection; (b) free string end and positive wave reflection.*

Boundary conditions impose constraints on the solution at the boundary of its domain. As an example, if y represents the displacement of a vibrating string boundary conditions impose values for y and its derivatives at the boundary points $x = 0$ and $x = L$. The two most common boundary conditions for a string are the *fixed end* condition and the *free end* condition, which read as follows for the boundary point $x = 0$ (similar equations can be written for the boundary point $x = L$)

$$\begin{aligned} &\text{Fixed end:} && y(x,t)|_{x=0} = 0, && \Rightarrow \quad y^+(ct) = -y^-(ct); \\ &\text{Free end:} && \left.\frac{\partial y}{\partial x}(x,t)\right|_{x=0} = 0, && \Rightarrow \quad y^+(ct) = y^-(ct). \end{aligned} \tag{3.17}$$

These equations show that boundary conditions imply "reflection" conditions on the traveling waves $y^{\pm}$ (see Fig. 3.3).

3.2.2.3 Waves and modes

A different analysis of the wave equation was proposed by Fourier, who proved that the general solution to Eq. (3.12) can be regarded as a superposition of a numerable set of so-called *stationary waves*.

We exemplify the Fourier analysis in the case of an ideal string with fixed-end boundary conditions $y(x,t)|_{x=0,L} \equiv 0$. We search for particular solutions $y(x,t) = s(x)q(t)$, which we call stationary waves, since they have a shape in space that is determined by the spatial function $s(x)$ and is modulated in time by the temporal function $q(t)$.

By substituting the generic stationary wave solution into Eq. (3.12), one finds that the functions s and q must satisfy suitable differential equations:

$$\frac{s''}{s}(x) = \frac{1}{c^2}\frac{\ddot{q}}{q}(t) \qquad \Rightarrow \quad s''(x) = \alpha s(x), \quad \ddot{q}(t) = c^2 \alpha q(t), \tag{3.18}$$

for some $\alpha \in \mathbb{R}$. This last equation follows from the fact that s''/s is a function of space only, while $\ddot{q}/q$ is a function of time only. Therefore these ratios must necessarily equal to a constant α.

Now look at the spatial equation. In order for the boundary conditions to be satisfied s has necessarily to be a non-monotonic function and consequently the condition $\alpha < 0$ must hold, so that s obeys the

equation of a second-order oscillator (otherwise $s(x)$ would be an exponential function). Moreover, since it has to be $s(0) = s(L) = 0$, only a numerable set of spatial frequencies are allowed for s:

$$s(x) = \sqrt{\frac{2}{L}} \sin(k_n x), \qquad \text{with} \quad k_n = \frac{n\pi}{L}, \tag{3.19}$$

where $\sqrt{2/L}$ is just a normalization factor.

Once the spatial equation has been solved, the temporal equation gives

$$q(t) = A \sin(\omega_n t + \phi), \qquad \text{with} \quad \omega_n = ck_n = \frac{n\pi c}{L}, \tag{3.20}$$

where A and ϕ depend on initial conditions. Again, only a numerable set of temporal frequencies $\omega_n = ck_n$ are allowed. Spatial and temporal frequencies are proportional to each other through the constant c.

In conclusion we have obtained the following stationary waves, or *normal modes*:

$$y_n(x,t) = \sqrt{\frac{2}{L}} \sin(\omega_n t + \phi_n) \sin(k_n x). \tag{3.21}$$

The general solution to Eq. (3.12) can be expressed as a linear combination of these modes:

$$y(x,t) = \sum_{n=1}^{+\infty} A_n y_n(x,t), \tag{3.22}$$

where $A_n,\ \phi_n$ are determined by the initial conditions. This latter equation re-states what we already know: a periodic signal, such as the one generated in an ideal string with ideal boundary conditions, can be expressed as a series of harmonically-related sinusoidal signals.

Note that the Fourier solution, expressed in term of normal modes, and the D'Alembert solution, expressed in terms of traveling waves, are equivalent. In fact a standing wave $y_n(x,t)$ can be viewed as a superposition of sinusoidal traveling waves. More precisely, using the Werner formulas[1] a standing wave can be written as

$$y_n(x,t) = \sqrt{\frac{1}{2L}} \left\{ \cos[k_n(ct - x) + \phi_n] - \cos[k_n(ct + x) + \phi_n] \right\}. \tag{3.23}$$

Therefore a standing wave *is* the sum of two sinusoidal waves $y^{\pm}$ that translate rigidly with velocity c, in the right-going and left-going directions, respectively. This proves the equivalence of the D'Alemebert and Fourier solutions.

Note however that normal-mode solutions are more general than traveling-wave solutions: already a simple system like a one-dimensional bar, described by a 4th order PDE, does not admit a solution in terms of traveling waves while its normal modes can be written analytically.

3.3 Delays and oscillations

3.3.1 The Karplus-Strong algorithm

This section reviews a sound synthesis algorithm which is relevant from many viewpoints. First, the Karplus-Strong (KS hereafter) algorithm is a famous one and deserves to be studied. Second, it contains many of the basic elements that are needed to provide a clear picture of what waveguide modeling is

[1] $2 \sin\alpha \sin\beta = \cos(\alpha - \beta) - \cos(\alpha + \beta)$.

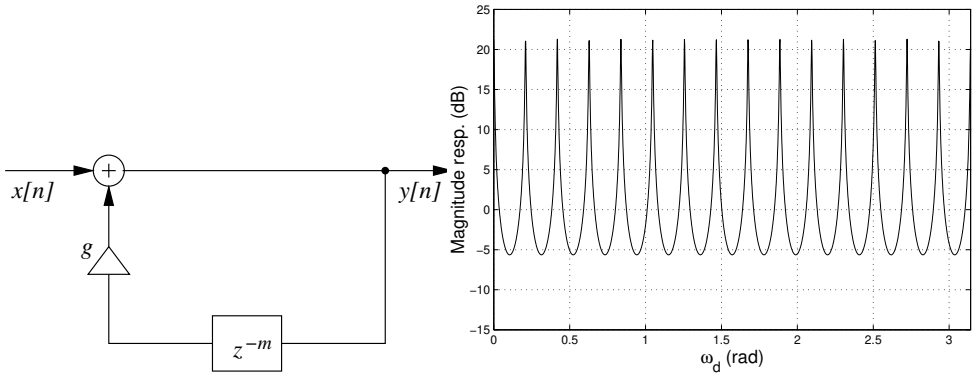


Figure 3.4: *A comb filter; (a) block scheme and (b) magnitude response.*

all about, and yet it is structurally simple enough to be discussed in a limited amount of pages. Finally, from a historical perspective it can be regarded as the first prototype of a waveguide approach to string modeling: it is true that the original formulation of the algorithm did not contain any physical interpretation. What is unquestionable, however, is that the KS algorithm is structurally identical to the simplest waveguide models that we are going to examine in the next sections.

3.3.1.1 The comb filter

The basic computational structure underlying the KS algorithm is the *IIR comb* filter, which is represented by the following difference equation (and transfer function):

$$y[n] = x[n] + gy[n-m], \quad \Rightarrow \quad H(z) = \frac{1}{1-gz^{-m}}, \tag{3.24}$$

with $g \in \mathbb{R}$. The block structure of the filter is depicted in Fig. 3.4(a). The poles of $H(z)$ are found from $z^m = g$. Therefore the filter has m poles $p_l = \sqrt[m]{g}e^{j2l\pi/m}$ for $l = 0, \dots m-1$, equally spaced around the circle of radius $\sqrt[m]{g}$. In order for the filter to be stable, the condition $\mid g \mid < 1$ must be satisfied.

The corresponding magnitude response is plotted in Fig. 3.4(b). Each pole p_l produces a peak in the magnitude response. We can apply the analysis seen in Chapter *Sound modeling: signal based approaches* for the second-order resonating filter in order to understand the relation between the poles p_l and the peaks in the response: in general, as g increases and grows closer to 1, each peak becomes higher and the associated bandwidth narrows down. Note also that the filter produces a harmonic spectrum in which frequency peaks are integer multiples of the "fundamental" frequency $f_0 = F_s/m$ Hz.

Figure 3.4(a) already provides us with an intuitive proto-physical interpretation: a disturbance (a wave) in a medium is propagated through that medium, is confined within a certain length, bounces back and forth due to some boundary conditions, has some energy dissipated at each bounce through the coefficient g. Note that if the sign of the wave is inverted at each reflection, the resulting filter spectrum is affected:

$$y[n] = x[n] - gy[n-m] \quad \Rightarrow \quad H(z) = \frac{1}{1+gz^{-m}} \tag{3.25}$$

In this case the poles are $p_l = \sqrt[m]{g}e^{j(2l+1)\pi/m}$ for $l = 0, \dots m-1$. This means that the corresponding frequency peaks have all been shifted by an angle π/m with respect to the previous case: now the frequency peaks are *odd* integer multiples of the "fundamental" frequency $f_0 = F_s/(2m)$ Hz. Section 3.4.3

will show that choosing a sign or another corresponds to describing two different boundary conditions (e.g., an open termination versus a closed termination in an acoustical bore).

M-3.1

Write a function that computes the output of the comb filter of Fig. 3.4, given a desired fundamental frequency f_0 and a factor $g < 1$.

M-3.1 Solution

```
function y = ks_simplecomb(f0,g);

global Fs;
m = round(Fs/f0);   %length of the delay line
d= dline_init(m);   % create a delay-line object

x=((rand(1, ceil(m))*2) -1)/2;% define random input vector of length m
x=[x zeros(1,round(-3*m/log10(g)) )]; % zero-pad x to hear sound tail
y=zeros(1, length(x));        % initialize output signal

for n = 1:length(x)  % audio cycle
    y(n) = x(n) + d.y*g; %read from delay line and update output
    d=dline_compute(d);  %update delay line
    d.x = y(n);
end
```

The input signal x is defined in accordance to the KS algorithm specifications (see next section). Zero padding of x is chosen in such a way that the ouput signal has time to decay by 60 dB (see Chapter *Sound in space*). Matlab/Octave are very inefficient at computing long cycles, but we use this approach for coherence with next examples; in particular we have used two auxiliary functions that initialize a delay line structure

```
function f = dline_init(d);  %initialize a dline structure of length d
% x is the current input value written into the line
% y=x(n-d) is the current output value read from the line
% in is a buffer containing d past input values

f.x = 0;   f.y = 0;
if(floor(d) == d)  f.d = d;  % ok, d is a valid integer delay
else               error('Not a valid delay');
end
f.in = zeros(1, d);  % create buffer for past input values
```

and update the state of a delay line structure

```
function f = dline_compute(f);

f.y = f.in(1);         % output is the first sample in the buffer
f.in = [f.in(2:length(f.in)), f.x];              % update buffer
```

3.3.1.2 Synthesis of plucked strings

The above observations suggest that the comb structure (3.24) may be employed to synthesize harmonic sounds, such as those produces by an ideal string. However, in order to obtain a complete formulation of the KS algorithm we still have to add some refinements to the structure. Specifically, what it is missing is a mean to control the spectral tilt of the filter magnitude response (i.e. the rate at which the response decays with increasing frequency), and to account for different decay rates for the sound partials.

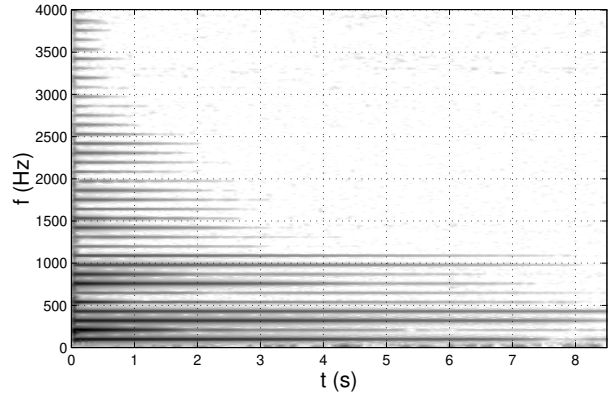


Figure 3.5: . *Spectrogram of a plucked A2 guitar string. Note the harmonic structure and the decay rates, which increases with increasing frequency.*

In the real world a nylon guitar string is one of the closest relative of an ideal string and exhibits an almost perfectly harmonic spectrum. Figure 3.5 shows the spectrogram of a a plucked guitar string: as expected, a harmonic spectrum can be observed. However another relevant feature is that each harmonic partial decays at a different rate, with lower partials surviving longer than higher partials.

On the other hand we have just seen that the IIR comb filter produces a spectrum in which all harmonic peaks have the same magnitude, which means that the associated partials all decay in time at the same rate. In order to simulate a frequency-dependent decay, one can insert a low-pass filter H_{lp} into the feedback loop, as shown in Fig. 3.6(a): we call this structure a *low-pass comb* filter. Intuitively, at each passage the high-frequency components are attenuated more strongly than low-frequencies components. The simplest low-pass filter that can be employed is the first-order FIR already examined in Chapter *Fundamentals of digital audio processing*:

$$y[n] = \frac{1}{2}\left[x[n] + x[n-1]\right] \quad \Rightarrow \quad H_{lp}(z) = \frac{1}{2}\left[1 + z^{-1}\right]. \tag{3.26}$$

Figure 3.6(b) shows the frequency response of the low-pass comb structure after the insertion of H_{lp}: as expected higher resonances are less peaked and have larger bandwidths, because now the filter poles have frequency-dependent magnitudes.

However the insertion on a low-pass filter in the structure has also a second effect: it introduces an additional half-sample delay, which can be observed if one looks at the phase response of $H_{lp}(z)$ and is qualitatively explained by the fact that this is filter averages the current sample with the previous one. A consequence of this additional delay is that the fundamental frequency generated by the low-pass comb structure is now $f_0 = F_s/(m + 1/2)$ Hz. Moreover, a closer analysis would also show that the upper partials are not anymore integer multiples of $f_0 = F_s/(m+1/2)$, due to the insertion of H_{lp} in the loop. These deviations from the harmonic series can also be noticed from the plot in Fig. 3.6(b).

In many cases the deviations introduced by the low-pass filter are very small, especially for the lower partials and for values of g that are close to 1. However they can still be perceivable. As an example, if $F_s = 44.1$ kHz and $m = 100$, then a half sample delay corresponds to a delay in the order of 10^{-5} s: in this case the IIR comb produces a fundamental at $F_s/m = 441$ Hz, while the low-pass comb produces a fundamental at $F_s/(m + 1/2) \sim 439$ Hz.

M-3.2

Find the response of the complete system given in Fig. 3.6 and plot magnitude and phase responses for various values of g and m.

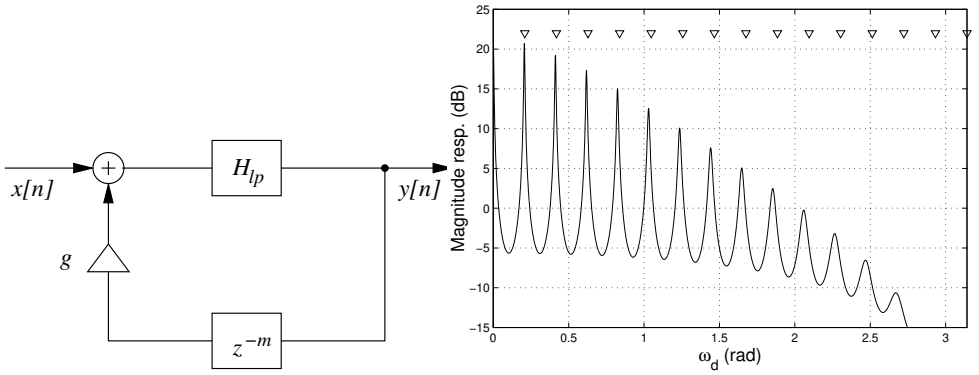


Figure 3.6: *Low-pass comb filter obtained through insertion of a low-pass element into the comb structure; (a) block scheme and (b) frequency response (the triangles mark the harmonic series $l\pi/L$, $l \in \mathbb{N}$).*

The low-pass comb structure discussed so far is the core of the KS algorithm. However we have not yet discussed what is the input signal to fed to the filter in order to obtain an output sound. Since the impulse response of the filter is the signal that is resemblant of a plucked string sound, an obvious choice is to inject the filter with an impulse. A second possible choice, originally suggested by Karplus and Strong, is to impose a random initial state (m past values of y) to the filter: although this choice has hardly any physical interpretation,[2] it has the benefit of providing significant initial excitation in the high-frequency region, with a consequent perceptual effect of an initial noisy transient followed by a harmonic steady-state signal.

M-3.3

Write a function that implements the KS algorithm using the low-pass comb of Fig. 3.6, given a desired fundamental frequency f_0 and a factor $g < 1$.

M-3.1 Solution

```
function y = ks_lpcomb(f0,g);

global Fs;
m = round(Fs/f0);   %length of the delay line
d= dline_init(m);   % create a delay-line object

x=((rand(1, ceil(m))*2) -1)/2;% define random input vector of length m
x=[x zeros(1,round(-3*m/log10(g)) )]; % zero-pad x to hear sound tail
y=zeros(1, length(x));        % initialize output signal
a_past=0;   %initialize auxiliary variable (input to lowpass filter)

for n = 1:length(x)  % audio cycle
    a = x(n) + d.y*g; %read from delay line and sum to input
    y(n) = 1/2 * (a + a_past); %update output through lowpass filter
    a_past = a; % update auxiliary variable
    d=dline_compute(d);  %update delay line
    d.x = y(n);
end
```

[2] It would be like imposing initial random displacements to points of a string, as we shall see in the next sections.

3.3.2 Fine tuning and fractional delays

Sound pitch (which we assume to coincide with fundamental frequency)[3] in the KS algorithm is quantized: adding a unit delay in the comb filter modifies the fundamental period by $1/F_s$, which is a rather gross and perceivable quantization. In order to obtain a finer tuning of the delay loop, we need techniques to simulate *fractional delays*.

An ideal delay of m samples is a filter with transfer function $H_m(z) = z^{-m}$. Therefore its frequency, magnitude, and phase responses are

$$H_m\left(e^{j\omega_d}\right) = e^{-j\omega_d m}, \quad \left|H_m\left(e^{j\omega_d}\right)\right| \equiv 1, \quad \arg\left[H_m\left(e^{j\omega_d}\right)\right] = -m\omega_d. \tag{3.27}$$

We want to design a filter with the same characteristics, i.e. flat magnitude response and linear phase response (equivalently, with constant and coincident phase and group delays). However we want the slope of the phase response to be an arbitrary phase delay τ_{ph}, and not limited to integer values m. Moreover, since any real delay τ_{ph} can be written as the sum of an integer delay $\lfloor \tau_{\text{ph}} \rfloor$ and and a *fractional delay* $0 \leq (\tau_{\text{ph}} - \lfloor \tau_{\text{ph}} \rfloor) < 1$, without loss of generality we restrict our attention to the design of fractional-delay filters $H_{\tau_{\text{ph}}}$ with $0 \leq \tau_{\text{ph}} < 1$).

Note that the impulse response of an ideal delay filter is

$$h_{\tau_{\text{ph}}}[n] = \frac{1}{2\pi} \int_{-\pi}^{+\pi} e^{-j\omega_d \tau_{\text{ph}}} e^{j\omega_d n} d\omega_d = \text{sinc}(n - \tau_{\text{ph}}). \tag{3.28}$$

If $\tau_{\text{ph}} = m \in \mathbb{N}$ this reduces to $h[n] = \delta[n - m]$. However, if τ_{ph} is not integer then this is a non-causal filter with infinite impulse response, i.e. a non-realizable filter. This remark makes it clear that we will not be able to find exact realizations of fractional-delay filters, and we will have to look for approximations.

3.3.2.1 FIR fractional delay filters

We first examine FIR fractional-delay filters, of the form

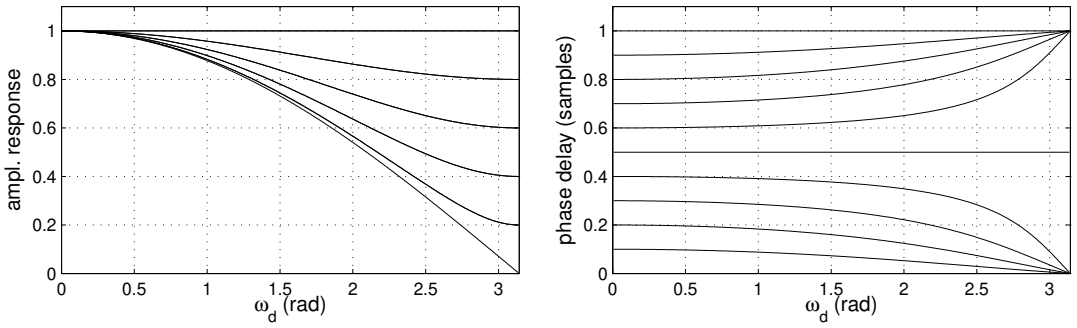
$$H_{\tau_{\text{ph}}}(z) = \sum_{k=0}^{N} b_k z^{-k}. \tag{3.29}$$

Starting from this general form, we have to design of an Nth order FIR filter approximating a constant magnitude and linear phase frequency response. Several criteria can be adopted to drive this approximation problem. One approach amounts to minimizing some error distance between the FIR filter (3.29) and the ideal fractional-delay filter defined previously. Possibly the most intuitive realization of this approach is the minimization of the least squared (LS) error function, defined as the L^2 norm the error frequency response $E(e^{j\omega_d}) = H_{\tau_{\text{ph}}}(e^{j\omega_d}) - e^{-j\tau_{\text{ph}}}$ (i.e. E is the difference between the frequency responses of the FIR filter and the ideal fractional-delay filter).

A different approach, that we describe in some more details, amounts to setting the error function $E(e^{j\omega_d})$ and its N derivatives to zero at $\omega_d = 0$:

$$\left.\frac{d^l E}{d\omega_d^l}(e^{j\omega_d})\right|_{\omega_d=0}, \qquad l = 0, \ldots N. \tag{3.30}$$

[3] In chapter *Auditory based processing* we will see that pitch perception is a complex phenomenon, and that the perceived pitch does not necessarily coincide with the fundamental frequency.



(a) (b)

Figure 3.7: *Linear interpolation filters ($N = 1$) for $\tau_{ph} = 0, 0.1, \ldots, 1$; (a) amplitude response and (b) phase delay.*

This is called the *maximally flat* design at $\omega_d = 0$, since it tries to make the error function as flat as possible around the value 0, in the vicinity of zero frequency. Substituting Eq. (3.29) in these latter $N + 1$ equations yields

$$\sum_{k=0}^{N} k^l b_k = \tau_{\text{ph}}^l, \quad l = 0, \ldots N \quad \Leftrightarrow \quad \boldsymbol{V}\boldsymbol{b} = \boldsymbol{\tau}, \tag{3.31}$$

where $\boldsymbol{b} = [b_0, b_1, \ldots, b_N]$, $\boldsymbol{\tau} = [1, \tau_{\text{ph}}, \ldots \tau_{\text{ph}}^N]$, and $\boldsymbol{V}$ is a Vandermonde matrix with elements $v_{i,j} = (j-1)^{i-1}$. Since $\boldsymbol{V}$ is non-singular, the system has a unique solution which can be written in explicit form as

$$b_k = \prod_{l \neq k; l=0}^{N} \frac{\tau_{\text{ph}} - l}{k - l}, \quad k = 0, \ldots N. \tag{3.32}$$

It is interesting to notice that the FIR filter coefficients obtained by this method are equal to those of the *Lagrange interpolation* formula for equally spaced abscissas. In other words, the FIR filter determined by these coefficients estimates the value $x[n - \tau_{\text{ph}}]$ by interpolating a polynomial of order N over the $N+1$ values $x[n], x[n-1], \ldots x[n-N]$. This leads to Lagrange interpolation.[4] For $N = 1$ one obtains simple linear interpolation, $b_0 = 1 - \tau_{\text{ph}},\ b_1 = \tau_{\text{ph}}$. For the case $\tau_{\text{ph}} = 1/2$ we reobtain the first-order FIR low-pass filter.

Plots for $N = 1$ and different values of τ_{ph} are shown in Fig. 3.7. The phase delay remains reasonably constant up to high frequency values (and is exactly constant in the cases $\tau_{\text{ph}} = 0, 1/2, 1$). Note however that the magnitude response has always a low-pass character. This is a drawback of these FIR filters: high frequencies are attenuated due to non flat magnitude response. Using higher orders N allows to keep the magnitude response close to unity and a phase response close to linear in a wider frequency band. Of course, this is paid in terms of computational complexity.

M-3.4

Implement a fractional delay line using Lagrange interpolation.

M-3.4 Solution

[4] We are not interested here in deriving the Lagrange interpolation method, which is reviewed in many textbooks of numerical analysis.

Same approach as before. One function to initialize the line

```
function f = lagrangedline_init(d,N); %uses Nth order lagrange interpolation

f.x = 0; f.y = 0;
f.d = d;            % set delay (not necessarily integer)
f.in = zeros(1, floor(d)+2); % create buffer for past input values
f.b=ones(1,N+1);       %coefficients of the Lagrange interpolator
tau=d-floor(d);       %fractional delay to be simulated
for k=1:length(f.b)
    for l=1:length(f.b)
        if (l~=k); f.b(k)=f.b(k)*(tau-(l-1))/( (k-1)-(l-1)); end
    end
end
```

and one to update the state

```
function f = lagrangedline_compute(f);

f.y = f.b * f.in(1:length(f.b))'; % output is lagrange interpolation of buffer
f.in = [f.in(2:length(f.in)), f.x]; % update buffer
```

These functions can be tested in the KS algorithm (examples M-3.1 and M-3.3) in place of the integer delay lines.

3.3.2.2 All-pass fractional delay filters

We now examine IIR fractional-delay filters, of the form

$$H_{\tau_{\text{ph}}}(z) = \frac{z^{-N}A(z)}{A(z^{-1})} = \frac{a_n + a_{n-1}z^{-1} + \ldots + a_1 z^{-(N-1)} + z^{-N}}{1 + a_1 z^{-1} + \ldots + a_{N-1}z^{-(N-1)} + a_n z^{-N}}. \tag{3.33}$$

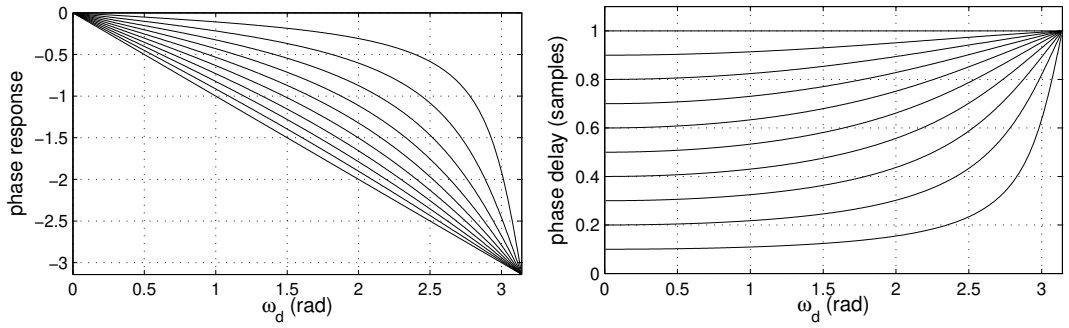
This is not the transfer function of a generic IIR filter. It represents the transfer function of a Nth order *all-pass* filter. According to the definition already given in Chapter *Fundamentals of digital audio processing*, an all-pass filter is a filter with a perfectly flat magnitude response. The filter in the above equation satisfies the property $\left| H_{\tau_{\text{ph}}}(e^{j\omega_d}) \right| \equiv 1$ by construction: this property can be proved by noting that, since the numerator polynomial is a mirrored version of the denominator polynomial A, the poles of a stable all-pass filter are located inside the unit circle and its zeros are located outside the unit circle with the same angle and with the inverse radius of the corresponding poles.

Since the above IIR filter satisfies by construction one of the two properties of an ideal delay filter (flat magnitude response), we can now focus on the second one (linear phase response). The phase response of an all-pass filter is found to be

$$\arg\left[H_{\tau_{\text{ph}}}\left(e^{j\omega_d}\right)\right] = N\omega_d + 2\arg\left[\frac{1}{A\left(e^{-j\omega_d}\right)}\right] = N\omega_d + 2\arctan\left[\frac{\sum_{k=0}^{N} a_k \sin(k\omega_d)}{\sum_{k=0}^{N} a_k \cos(k\omega_d)}\right]. \tag{3.34}$$

Therefore the phase response, the phase delay, and the group delay are all highly non-linear functions of the filter coefficients. This means that one cannot expect as simple design formulas for the all-pass filter coefficients as for FIR filters. Instead, one can almost exclusively find only iterative optimization techniques for minimization of traditional error criteria.

Possibly the only design technique that has a closed-form solution is the *maximally flat group delay* design. Let us start considering an all-pole low-pass filter with transfer function $1/A(z^{-1})$. It has been



(a) (b)

Figure 3.8: *First-order Thiran allpass filters for $\tau_{ph} = 0, 0.1, \ldots, 1$; (a) phase response and (b) phase delay.*

shown that the condition of maximally flat group delay at $\omega_d = 0$ for this filter yields the following analytic solution:

$$a_k = (-1)^k \binom{N}{k} \prod_{l=0}^{N} \frac{2\tau_{\text{ph}} + l}{2\tau_{\text{ph}} + k + l}, \tag{3.35}$$

where $\binom{N}{k}$ is the binomial coefficient. When $\tau_{\text{ph}} > 0$ then the filter is stable. This result can be applied to our problem using Eq. (3.34): since the fractional phase delay of $H_{\tau_{\text{ph}}}$ is twice those of $1/A$, a maximally flat all-pass filter with coefficients

$$a_k = (-1)^k \binom{N}{k} \prod_{l=0}^{N} \frac{\tau_{\text{ph}} + l}{\tau_{\text{ph}} + k + l}, \tag{3.36}$$

approximates the ideal delay filter with total delay $N + \tau_{\text{ph}}$. This is known as *Thiran all-pass filter* approximation.

As an example let us look at the first-order all-pass filter

$$H_{\tau_{\text{ph}}}(z) = \frac{a_1 + z^{-1}}{1 + a_1 z^{-1}}, \tag{3.37}$$

with $a_1 < 1$ for stability. The plots of its phase response and phase delay are shown in Fig. 3.8. In the low-frequency region, the phase response can be approximated as follows:

$$\arg\left[H_{\tau_{\text{ph}}}\left(e^{j\omega_d}\right)\right] \sim -\frac{\sin\omega_d}{a_1 + \cos(\omega_d)} + \frac{a_1 \sin\omega_d}{1 + a_1 \cos\omega_d} \sim -\omega_d \frac{1 - a_1}{1 + a_1}, \tag{3.38}$$

i.e. the phase response is approximately linear with phase and group delay approximately equal to $(1 - a_1)/(1 + a_1)$. Therefore given a desired phase delay τ_{ph} one chooses

$$a_1 = \frac{1 - \tau_{\text{ph}}}{1 + \tau_{\text{ph}}}. \tag{3.39}$$

This corresponds to the Thiran approximation with $N = 1$.

Thiran filters have complementary drawbacks with respect to Lagrange filters: although they provide flat magnitude response, detuning of higher frequencies occurs due to phase non-linearity. In order to have phase response approximately linear in a wider frequency range one has to use higher orders, at the expense of higher complexities.

M-3.5

Implement a fractional delay line using Thiran filters.

M-3.5 Solution

Same approach as before. One function to initialize the line

```
function f = thirandline_init(d, N); %uses a Nth order Thiran filter

f.x = 0; f.y = 0;
f.d = d;            % set delay (not necessarily integer)
f.in = zeros(1, floor(d)-N); % create buffer for past input values
       % the Thiran filter account for the remaining N+(d-floor(d))
f.state=zeros(1,N);   %state of the Thiran filter
f.a = zeros(1, N+1); % coefficients of the Thiran filter
tau = d-floor(d);   %fractional delay to be simulated
for k = 0:N
    ak = 1;   for l=0:N; ak = ak * (tau+l)/(tau+k+l); end
    f.a(k+1) = (-1)^k * nchoosek(N,k) * ak;
end
```

and one to update the state

```
function f = thirandline_compute(f);

[out,state] = filter(fliplr(f.a), f.a, f.in(1), f.state);
f.state=state;
f.y=out;
f.in = [f.in(2:length(f.in)), f.x];
```

These functions can be tested in the KS algorithm (examples M-3.1 and M-3.3) in place of the integer delay lines.

3.3.2.3 Time-varying delays

3.4 Distributed models: the waveguide approach

This section introduces the basic concepts of waveguide modeling. Discussion is focused on one-dimensional resonators, and no attention is devoted here to higher dimensional waveguide structures.

In their simplest form, waveguide models exploit the existence of an analytical solution to the D'Alembert wave equation, which can be seen as a superposition of traveling waves (rigidly translating waveforms). Such a solution can be simulated in the discrete space-temporal domain using delay lines, and the resulting numerical algorithms are extremely efficient and accurate. Moreover, physical phenomena such as frequency dependent losses and dispersion can be included in the models by incorporating low-pass and all-pass filters in the delay line scheme. Again, careful design of such filters allows for very accurate and relatively low-cost simulations.

3.4.1 Basic waveguide structures

3.4.1.1 Wave variables and wave impedance

So far, only displacement y (for a string) and acoustic pressure p (for a cylindrical bore) have been considered in the wave equation. However, alternative wave variables can be used in strings and acoustical

bores. As an example, the force acting on a string section dx is defined as

$$f(x,t) = -T\frac{\partial y}{\partial x}(x,t) = -T\left[\frac{\partial y^+}{\partial x}(ct-x) + \frac{\partial y^-}{\partial x}(ct+x)\right] = \frac{T}{c}\left[\dot{y}^+(ct-x) - \dot{y}^-(ct+x)\right]. \quad (3.40)$$

Therefore, using this equation force waves $f^\pm$ can be defined as $f^\pm := \mp\frac{T}{c}\dot{y}^\pm$. On the other hand, the transversal velocity in the same string is given by

$$v(x,t) = \frac{\partial y}{\partial t}(x,t) = \dot{y}^+(ct-x) + \dot{y}^-(ct+x). \quad (3.41)$$

From this, velocity waves $v^\pm$ are defined as $v^\pm := \dot{y}^\pm$. As we have seen in Sec. 3.2.1, the force-velocity variable pair represent the mechanical Kirchhoff variables, in analogy with voltage and current in electrical systems. From the previous equations it immediately follows that

$$f^\pm(ct \mp x) = \pm Z_0 v^\pm(ct \mp x), \qquad \text{with} \qquad Z_0 = T/c = \sqrt{T\mu}. \quad (3.42)$$

The quantity Z_0 takes the name of *wave* (or *characteristic*) *impedance* of the string, and its reciprocal $\Gamma_0 = Z_0^{-1}$ is termed *wave admittance*. Note that using Z_0 both the force f and the velocity v can be related to the force waves $f^\pm$. Namely, the following relations hold:

$$\begin{aligned} f &= f^+ + f^-, & v &= \frac{1}{Z_0}\left[f^+ - f^-\right], \\ f^+ &= \frac{f + Z_0 v}{2}, & f^- &= \frac{f - Z_0 v}{2}, \end{aligned} \quad (3.43)$$

that transform the pair (f, v) into the pair (f^+, f^-), and vice versa.

Wave impedance can be defined also in a cylindrical bore. In this case the Kirchhoff variables are taken to be pressure p and flow u (volume velocity). These can be related through the wave impedance Z_0: $p^\pm(ct \pm x) = \pm Z_0 u^\pm(ct \pm x)$, where $Z_0 = \rho_{air}c/S$ and S is the constant cross-sectional area of the bore. For conical geometries, the cross-section S is not constant and the definition of Z_0 has to be generalized. The wave impedance is then defined as a function $Z_0(s)$ such that the relations $P^\pm(r,s) = \pm Z_0(s)U^\pm(r,s)$ hold in the Laplace domain. It can be seen that $Z_0(s) = \rho_{air}c/S \cdot [rs/(rs+c)]$.

In summary, Kirchhoff and wave variables in elastic media obeying the D'Alembert equation are related through wave impedance and Eqs. (3.43). This results provide the basis for developing 1-D waveguide structures.

3.4.1.2 Delay lines

Waveguide models exploit the existence of the solution (3.15) to the D'Alembert equation and discretize this solution instead of the differential equation itself. This remark explains to a large extent why waveguide structures are much more efficient than finite difference methods in simulating vibrations of elastic media, at least in the 1-D case.

As a starting example, consider a pressure distribution $p = p^+ + p^-$ inside an ideal lossless cylindrical bore. We want to discretize p both in time and in space. If T_s is the sampling period, a suitable choice for the spatial sampling step is $X_s = cT_s$. Assume for simplicity that the length L is a multiple of the spatial step, $L = mX_s$. Then a discretized version of p is obtained through the variable substitution $x \mapsto mX_s$ and $t \mapsto nT_s$ (with $m, n \in \mathbb{N}$), and leads to

$$p(mX_s, nT_s) = p^+(ncT_s - mX_s) + p^-(ncT_s + mX_s) = p^+((n-m)cT_s) + p^-((n+m)cT_s).$$

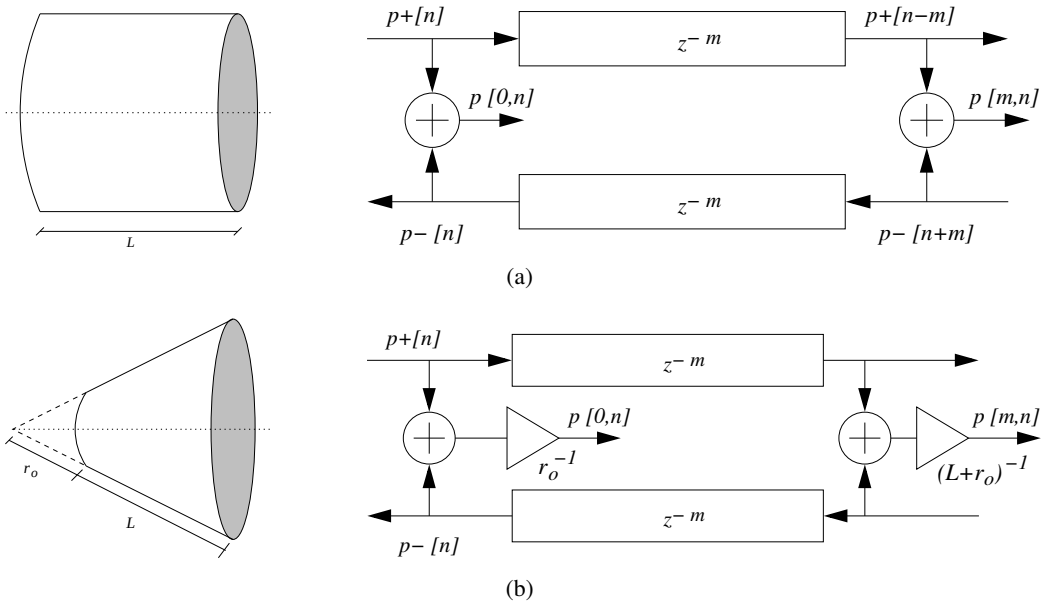


Figure 3.9: *Lossless waveguide sections with observation points at position $x = 0$ and $x = mX_s = L$; (a) cylindrical section; (b) conical section.*

Removing the constant sampling steps yields

$$p[m, n] = p^+[n - m] + p^-[n + m]. \tag{3.44}$$

The term $p^+[n - m]$ in Eq. (3.44) can be thought of as the output from a digital delay line of length m, whose input is $p^+[n]$. Analogously, the term $p^-[n + m]$ can be thought of as the input of a digital delay line with the same length, whose output is $p^-[n]$. This remark leads to the definition of a *waveguide section* as a bidirectional delay line, as depicted in Fig. 3.9(a). The horizontal direction of this structure has a straightforward physical interpretation: it corresponds to the position x along the axis of the cylindrical bore. In the example depicted in Fig. 3.9(a), two "observation points" have been chosen at $x = 0$ and $x = mX_s = L$. At these points, the pressure signal at time n is reconstructed by summing the corresponding pressure waves $p^{\pm}$.

A very similar structure can be outlined for numerically simulating a pressure distribution in an ideal lossless conical bore. In this case, propagation is described by the one-dimensional equation (3.13), whose general solution is given by

$$R(r, t) = \frac{1}{r}[\tilde{R}^+(ct - r) + \tilde{R}^-(ct + r)]. \tag{3.45}$$

The conical waveguide is therefore defined as in Fig. 3.9(b). Observation points can be chosen analogously to the cylindrical case.

At the beginning of this discussion we have assumed for simplicity that $L = mX_s$. However this quantization of the allowed lengths is too coarse for our purposes: with a sampling rate $F_s = 44.1$ kHz and with a wave velocity $c = 347$ m/s (sound velocity in air at 20 C°), the resulting spatial step is $X_s = 7.8 \cdot 10^{-3}$ m. Length differences of this magnitude produce perceivable pitch variations in a wind instrument. One way to overcome this limitation is to include in the structure a fractional-delay filter (see Sec. 3.3.2) that provide fine tuning of the length of a waveguide section.

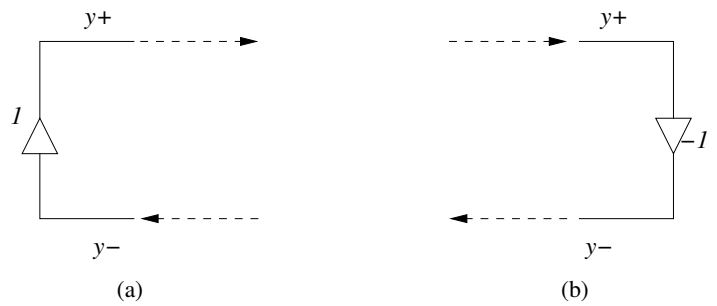


Figure 3.10: *Ideal waveguide terminations: (a) positive reflection; (b) negative reflection.*

3.4.1.3 Boundary conditions

Looking at Fig. 3.9 we immediately realize that we still one element in order to come out with a computational structure that describes e.g. a string with fixed ends or a cylindrical tube section with open ends: boundary conditions.

In Sec. 3.2.2 we have briefly discussed fixed-end and free-end boundary conditions for the displacement $y(x,t)|_{x=0,L}$ of a vibrating string. These can be immediately turned into *reflection conditions* for both velocity waves and force waves. As an example, a fixed-end condition implies that the velocity is 0 at the boundaries, therefore the reflection conditions $v^+ = -v^-$ applies at both points. By looking at Eq. (3.43), one also see that the 0 velocity condition translates into the reflection condition $f^+ = f^-$ at both points. Therefore wave variables at the boundaries are multiplied by either 1 or -1 (see Fig. 3.10).

More in general, reflection conditions can be derived by formulating boundary conditions for Kirchhoff variables and then using Eq. (3.43) to relate Kirchhoff variables to wave variables. A second relevant example is that of a cylindrical bore of length L, with a closed end at $x = 0$ and an open end at $x = L$. The first condition implies $u = u^+ + u^- = [p^+ - p^-]/Z_0 = 0$ at $x = 0$ (no flow through a closed end), which in turn implies the reflection conditions $u^+ = -u^-$ and $p^+ = p^-$. The second condition implies $p = p^+ + p^- = 0$ at $x = L$ (p matches the atmospheric pressure at the open boundary), which in turn implies the reflection conditions $p^- = -p^+$ and $u^+ = u^-$.

With these concepts in mind we can now go back to Sec. 3.3.1 and reinterpret the IIR comb structure used to construct the KS algorithm. The IIR comb can be viewed as a pair of waveguide sections of length $m/2$ samples in which traveling waves circulate and reflect at the boundaries according to some reflection condition. If the coefficient g has a positive sign, as in Eq. (3.24), the corresponding condition is that of a string fixed at both ends. The signal traveling into the filter can be interpreted either as a velocity wave (two sign inversions at the boundaries) or as a force wave (no sign inversions at the boundaries). As a result a harmonic spectrum is generated that contains all the partials. On the other hand, if the coefficient g has a negative sign, as in Eq. (3.25), the corresponding condition is e.g. that of a cylindrical bore with one open end and one closed end. The signal traveling into the filter can be interpreted either as a flow wave or as a pressure wave (both with one sign inversion at the boundaries). As a result a harmonic spectrum is generated that contains only the odd partials.

3.4.2 Modeling real world phenomena

As already mentioned, the waveguide structures introduced above describe *ideal* systems, i.e. ideally elastic media, where the D'Alembert equation (3.12) or its spherical version (3.13) hold. Real systems exhibit more complex behaviors. Two phenomena are particularly relevant for sound production: dissi-

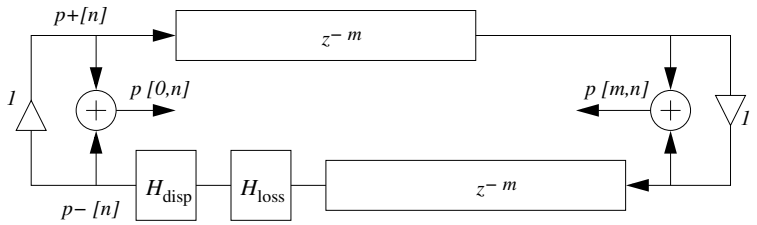


Figure 3.11: *Waveguide simulation dissipation and dispersion phenomena through insertion of loss and dispersion filters.*

pation and dispersion. Both can be accounted for by adding proper time, space or time-space derivatives of different orders to the ideal wave equation. Correspondingly the basic waveguide structure is modified by inserting appropriate loss and dispersion filters in the loop, as in Fig. 3.11

3.4.2.1 Dissipation

Energy *dissipation* occurs in any real vibrating medium. In an acoustical bore this is due to air viscosity, thermal conduction and wall losses. Dissipation in a string comes from internal losses related to elastic properties of the material, energy transfer through terminations, and friction with air. For clarity, consider the pressure distribution in a cylindrical bore. In the simplest approximation, all of the dissipation phenomena can be incorporated in the D'Alembert equation by including an additional term proportional to the first time derivative. As an example, a first-order approximation of a string with linear density μ, tension T, and dissipation is given by the modified D'Alembert equation

$$\mu \frac{\partial^2 p}{\partial t^2}(x,t) = T \frac{\partial^2 p}{\partial x^2}(x,t) - d_1 \frac{\partial p}{\partial t}(x,t). \tag{3.46}$$

In the limit of small d_1, Eq. (3.46) still admits a traveling wave solution, which can be digitized with the same procedure described in the ideal case:

$$\begin{aligned} p(x,t) &= e^{-\frac{d_1 x}{2c}} p^+(ct-x) + e^{\frac{d_1 x}{2c}} p^-(ct+x), \quad \text{then} \\ p[m,n] &= g^m p^+[n-m] + g^{-m} p^-[n+m], \quad \text{with} \quad g = e^{-\frac{d_1 T_s}{2}} < 1. \end{aligned} \tag{3.47}$$

Thus the traveling waves are exponentially damped along the propagation direction, and this phenomenon can be incorporated in the waveguide structure. In many real-world phenomena, however, losses increase with frequency. As an example, the dissipative force exerted by the air on a moving string section is, to a first approximation, directly proportional to the frequency of oscillation. Similar remarks apply to the effects of internal material losses. A better approximation of dissipation phenomena in a string is provided by the equation

$$\mu \frac{\partial^2 p}{\partial t^2}(x,t) = T \frac{\partial^2 p}{\partial x^2}(x,t) - d_1 \frac{\partial p}{\partial t}(x,t) + d_2 \frac{\partial^3 p}{\partial t \partial^2 x}(x,t), \tag{3.48}$$

where d_1 introduces frequency-independent dissipation and d_2 introduces frequency-dependent dissipation. This frequency dependence can be accounted for by substituting the constant factor g with a loss filter, which will have a low-pass characteristics. This is shown in Fig. 3.11, where losses have been consolidated, or *lumped*, in a single loss filter $H_{\text{loss}}(z)$ cascaded to the delay line. This filter summarizes the distributed losses occurring in the spatial interval $[0, 2mX_s]$.

With these concepts in mind we can go back again to Sec. 3.3.1 and reinterpret the the comb structures. In the simple IIR comb filter, the coefficient $g < 1$ plays the role of the loss factor g^m, and accordingly introduces equal decay times to all partials. In the low-pass comb filter, the low-pass transfer function H_{lp} plays the role of the loss filter $H_{\text{loss}}(z)$, and accordingly introduces frequency-dependent decay times to the partials.

3.4.2.2 Loss filter design

There are many techniques for designing a loss filter H_{loss} to fit a real object. In this section we outline a relatively simple approach to fit a lossy waveguide model to a real string sound.

First the sound of the target string has to be recorded and analyzed. This can be done using e.g. the sinusoidal peak detection/continuation algorithms discussed in Chapter *Sound modeling: signal based approaches*. As a result from the analysis stage, the frequencies f_k and the decay times τ_k ($k = 1, \ldots, N$) of the first N partials can be estimated. In particular τ_k is defined as the time required by the amplitude of the kth partial to decay by $1/e$ with respect to its initial amplitude. A robust way of calculating the τ_k's is fitting a line by linear regression on the logarithm of the amplitude envelopes derived from the peak continuation algorithm.

The estimated parameters f_k, τ_k specify the magnitude of H_{loss} over a set of N points:

$$\left| H_{\text{loss}}\left(e^{j\frac{2\pi f_k}{F_s}}\right)\right| = e^{-\frac{k}{f_k \tau_k}}, \quad k = 1, \ldots, N. \tag{3.49}$$

Given this magnitude specification, a common technique to design H_{loss} is through minimization of the squared error $\sum_{k=1}^{N}(H_{\text{loss}}\left(e^{j2\pi f_k/F_s}\right) - e^{-k/f_k\tau_k})^2$. However one problem with these techniques is that one may find a filter whose magniture exceeds unity, which would result in an unstable waveguide structure. Moreover, in order to avoid frequency dependent delay, $H_{\text{loss}}(z)$ should be ideally a linear-phase filter (and the length of the delay line should be reduced correspondingly, in order to obtain the desired overall delay).

A more straightforward design approach amount to choose a first order IIR low-pass filter:

$$H_{\text{loss}}(z) = g\frac{1+\alpha}{1+\alpha z^{-1}}, \tag{3.50}$$

with $-1 < \alpha < 0$ and $g < 1$. One can show that in this case the approximate analytical formulas for the decay times are

$$\frac{1}{\tau_k} \simeq a + b\left(\frac{2\pi f_k}{F_s}\right)^2, \qquad \text{with} \quad a = f_0(1-g), \quad b = -f_0\frac{\alpha}{2(\alpha+1)}^2, \tag{3.51}$$

and where f_0 is the fundamental frequency. Therefore the decay rate $1/\tau_k$ is a second-order polynomial of f_k with even order terms. Consequently a and b can be straightforwardly determined by polynomial regression from the prescribed decay times, and finally g and α are computed from a and b via the inverse of Eqs. (3.51). In most cases, the one-pole loss filter yields good results. Nevertheless, when precise rendering of the partial envelopes is required, higher-order filters have to be used.

M-3.6

Realize a complete loss filter design procedure, to be applied to a guitar sound. Use the spectral analysis tools to estimate the decay times of the guitar string partials. Use Eq. (3.51) to design the filter (3.50).

3.4.2.3 Dispersion

A second important phenomenon in natural wave propagation is *dispersion*. In a string, dispersion is introduced by string stiffness, i.e. the phenomenon by which a string opposes resistance to bending. Such a shearing force can be modeled as a fourth spatial derivative, which is introduced as an additional term in the D'Alembert equation:

$$\mu \frac{\partial^2 p}{\partial t^2}(x,t) = T \frac{\partial^2 p}{\partial x^2}(x,t) - D \frac{\partial^4 p}{\partial^4 x}(x,t), \tag{3.52}$$

where the dispersive correction term D is usually termed "bending stiffness" of the string, and is proportional to the string Young's modulus. If D is sufficiently small, its first-order effect is to increase the wave propagation speed with frequency:

$$c(\omega) = c_0 \left(1 + \frac{D\omega^2}{2Tc_0^2}\right), \tag{3.53}$$

where $c_0 = \sqrt{T/\mu}$ is now the wave propagation speed in the absence of dispersion. Equation (3.53) states that a traveling wave is no longer a rigid shape that translate at constant speed. Instead, frequencies "disperse" as they propagate with different velocities.[5] As a consequence, the frequencies f_k of the allowed partials are not harmonic, instead they are stretched onto an inharmonic series according to the equation

$$f_k = k f_0 I_k, \qquad \text{where} \quad I_k \approx \sqrt{1 + Bk^2}, \tag{3.54}$$

and where $B = \pi^2 D/TL^2$. The quantity I_k is usually termed *index of inharmonicity*. Dispersion is particularly important in piano strings, where the lower tones exhibit significant inharmonicity.

Having a non-uniform wave velocity $c(\omega)$ implies that it is not possible to define a sampling step as $X_s = c_0 T_s$. Instead, it can be said that a component with frequency $f = \omega/(2\pi)$ travels a distance $c_0 T_s$ in the time interval $c_0 T_s / c(\omega)$. As a consequence, each unitary delay z^{-1} in the waveguide structure has to be substituted with an all-pass dispersion filter with unitary magnitude response and a non-linear phase response approximates the frequency-dependent phase delay $c_0 T_s / c(\omega)$.

Similarly to dissipative low-pass filters, these all-pass delays can be *lumped* into a single product filter. Moreover, the linear and non-linear parts of the phase response can be treated separately. In conclusion the dispersion filter that substitutes $2m$ unitary delays can be written as $z^{-2m} H_{\text{disp}}(z)$, where z^{-2m} accounts for the linear part of the phase response and the all-pass filter $H_{\text{disp}}(z)$ approximates the non-linear part. The resulting dispersive waveguide structure is then as in Fig. 3.11.

M-3.7

Implement the waveguide structure of Fig. 3.11, including the loss filter (3.50) and an all-pass filter to simulate dispersion.

3.4.2.4 Dispersion filter design

Similarly to the discussion on loss filter design, the effects of dispersion in a real sound can be estimated from analysis using e.g. the sinusoidal peak detection/continuation algorithms discussed in Chapter *Sound modeling: signal based approaches*. The estimated series of partial frequencies f_k provide an indication of the degree of inharmonicity in the sound, and thus of dispersion.

[5] Dispersion can be sometimes experienced when hiking on the mountains, by imparting an impulse on a long metallic cable such as that of a cableway: after some seconds the impulse will bounce back and one will feel that it has "unraveled" into a smoother step with high-frequency ripples running out ahead.

In this section we outline one possible approach to the dispersion filter design. The total phase delay over a waveguide loop of length $2m$, with loss and dispertion filters is

$$\tau_{\text{ph}}(f_k) = \frac{kF_s}{f_k} = 2m + \tau_{\text{loss}}(f_k) + \tau_{\text{disp}}(f_k). \tag{3.55}$$

With everything else known, this equation provide a phase delay specification for the dispersion filter:

$$\tau_{\text{disp}}(f_k) = \frac{kF_s}{f_k} - 2m - \tau_{\text{loss}}(f_k). \tag{3.56}$$

Given L estimated partial frequencies $\{f_k\}_{k=1,\ldots,L}$, one can then design an all-pass filter of order $N < L$ as follows. First, for each partial compute the quantities

$$\beta_k = -\frac{1}{2}\left[\tau_{\text{disp}}(f_k) - 2N\pi f_k\right], \qquad k = 1, \ldots, L. \tag{3.57}$$

Then, filter coefficients are computed by solving the system

$$\sum_{j=1}^{N} a_j \sin(\beta_k + 2j\pi f_k) = \sin\beta_k, \qquad k = 1, \ldots, L. \tag{3.58}$$

This is an overdetermined system. It can be solved with a LS error criterion.

Note that this design approach is not based on the fitting of the relative positioning of the partials, but on the absolute values of the f_k's. Therefore the resulting all-pass filter accounts both for the simulation of dispersion and for the fine-tuning of the string (fractional delay).

M-3.8

Realize a complete dispersion filter design procedure, to be applied to a piano sound. Use the spectral analysis tools to estimate the frequencies of the piano partials. Use Eq. (3.58) to design the all-pass filter.

3.4.3 Junctions and networks

The last section has introduced the main concepts of waveguide modeling for a signal propagating in a *uniform* medium. When discontinuities are encountered, the wave impedance changes and signal *scattering* occurs, i.e. a traveling wave is partially reflected and partially transmitted.

Examples of non-uniform media are a cylindrical bore where the cross-sectional area changes abruptly, or a string where the value of the linear mass density jumps changes discontinuously. In order to model these discontinuities, appropriate junctions have to be developed, that connect two (or more) waveguide sections. The boundary reflection conditions that we have examined at the end of Sec. 3.4.1 can be regarded as special cases of junctions, as discussed in the following paragraphs.

3.4.3.1 The Kelly-Lochbaum junction

Consider two cylindrical bores, with cross-sectional areas $S_{1,2}$ and wave admittances $\Gamma_{1,2} = Z_{1,2}^{-1} = S_{1,2}/\rho_{air}c$, connected to each other. Analysis of this problem leads to the derivation of the well known *Kelly-Lochbaum* junction.

The derivation is based on imposing appropriate physical constraints on the Kirchhoff variables p, u at the junction. Specifically, continuity requires that pressures $p_{1,2}$ have the same value p_J at the junction. Moreover, the flows $u_{1,2}$ from the two sides must sum to zero (simply said, the air entering one side of

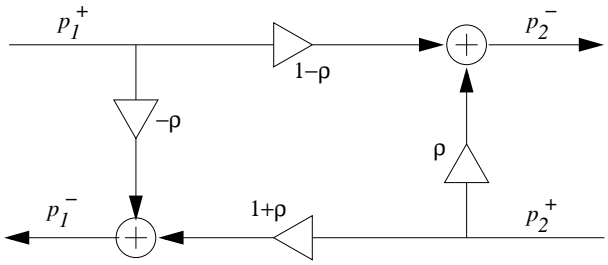


Figure 3.12: *Kelly-Lochbaum junction for two cylindrical bores with different areas.*

the junction and the air coming out from the other side must be the same). These two requirements lead to the following conditions at the junction:

$$u_1 + u_2 = 0, \qquad p_1 = p_2 = p_J. \tag{3.59}$$

Using the Kirchhoff analogy $p \leftrightarrow v$ (voltage) and $u \leftrightarrow i$ (current), Eqs. (3.59) can be regarded as describing a parallel junction. If pressure wave variables are introduced as in Eq. (3.43) (with p^+ and p^- denoting incoming and outgoing waves, respectively), and the junction pressure p_J is used, then the relation $p_l^- = p_J - p_l^+$ (for $l = 1, 2$) holds. Substitution in the first of Eqs. (3.59) yields

$$\begin{aligned} 0 &= (u_1^+ + u_1^-) + (u_2^+ + u_2^-) = \Gamma_1(p_1^+ - p_1^-) + \Gamma_2(p_2^+ - p_2^-) = \\ &= \Gamma_1(2p_1^+ - p_J) + \Gamma_2(2p_2^+ - p_J). \end{aligned} \tag{3.60}$$

From this, the junction pressure p_J can be expressed in terms of the incoming pressure waves $p_{1,2}^+$ as

$$p_J = 2\frac{\Gamma_1 p_1^+ + \Gamma_2 p_2^+}{\Gamma_1 + \Gamma_2}. \tag{3.61}$$

Using this latter expression, the outgoing pressure waves $p_{1,2}^-$ can be written as

$$\begin{aligned} p_1^- &= p_J - p_1^+ = -\frac{\Gamma_2 - \Gamma_1}{\Gamma_2 + \Gamma_1} p_1^+ + \frac{2\Gamma_2}{\Gamma_2 + \Gamma_1} p_2^+, \\ p_2^- &= p_J - p_2^+ = +\frac{2\Gamma_1}{\Gamma_2 + \Gamma_1} p_1^+ + \frac{\Gamma_2 - \Gamma_1}{\Gamma_2 + \Gamma_1} p_2^+. \end{aligned} \tag{3.62}$$

And finally

$$\begin{aligned} p_1^- &= -\rho\, p_1^+ + (1+\rho) p_2^+, \\ p_2^- &= (1-\rho) p_1^+ + \rho\, p_2^+, \end{aligned} \qquad \text{with} \quad \rho \triangleq \frac{\Gamma_2 - \Gamma_1}{\Gamma_2 + \Gamma_1}, \tag{3.63}$$

These equations describe the Kelly-Lochbaum junction. The quantity ρ is called the *reflection coefficient* of the junction. A scattering diagram is depicted in Fig. 3.12.

This junction has been extensively used in so-called "multitube lossless models" of the vocal tract. These are articulatory models where the vocal tract shape is approximated as a series of concatenated cylindrical sections. Pressure wave propagation in each section is then described using digital waveguides, and interconnections are treated as Kelly-Lochbaum junctions. However this very same junction can be used to describe not only acoustic, but also mechanical structures. As an example, consider two strings with different densities, connected at one point: this can be thought of as a series junction, since the physical constraints impose that velocity (i.e., "current") has to be the same on the left and right

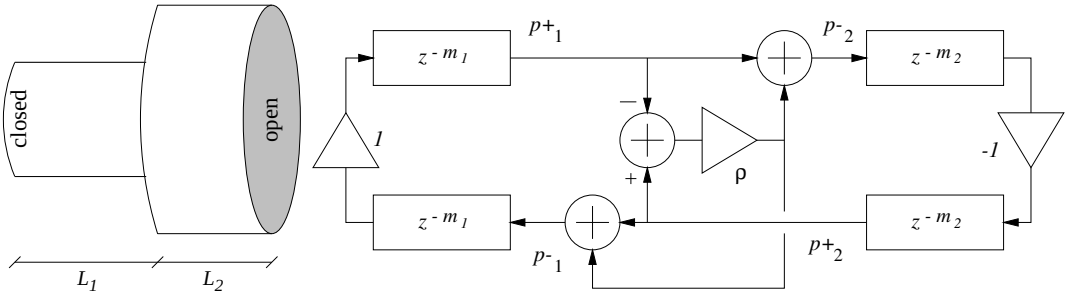


Figure 3.13: *Example of use of the Kelly-Lochbaum junction: (a) a parallel junction of two cylindrical bores; (b) realization with two waveguide sections and a Kelly-Lochbaum junction.*

sides, and the sum of forces (i.e., "voltages") from the two sides must be zero. Analogously to the above analysis, a series Kelly-Lochbaum junction can be derived in this case.

Terminations of a waveguide model are an interesting particular case of junctions. Consider an ideal cylindrical bore, closed at one end: this boundary condition corresponds to an infinite impedance $Z_2 = \infty$ (i.e., $S_2 = 0$), and thus to a reflection coefficient $\rho = -1$. In other words, complete reflection occurs and the relation $p_1^-(0,t) = p_1^+(0,t)$ holds. Similarly, an ideally open end can be seen to correspond to $Z_2 = 0$ (i.e., $S_2 = \infty$), and thus to $\rho = 1$: this is a second case where complete reflection occurs, namely the relation $p_1^-(0,t) = -p_1^+(0,t)$ holds. These reflection conditions are identical to those derived in Sec. 3.4.1 (similar considerations hold for string terminations).

Figure 3.13 shows an example where different junctions have been used and combined into a waveguide model. Note that in this example the scattering junction between the two cylindrical sections is not in the original Kelly-Lochbaum form; instead, a *one-multiply scattering junction* is used, which allows more efficient implementation of Eqs. (3.63). Open- and closed-tube terminations are modeled according to the above remarks.

M-3.9

Implement the waveguide structure of Fig. 3.13. Add a loss filter (3.50) to each WG section.

3.4.3.2 N-dimensional and loaded junctions

The result expressed in Eq. (3.63) can be readily extended to higher dimensions. Consider a parallel junction of N acoustic bores. In this case a *scattering matrix* can be found, and Eq. (3.63) is generalized to

$$\boldsymbol{p}^- = \boldsymbol{A} \cdot \boldsymbol{p}^+, \tag{3.64}$$

where $\boldsymbol{p}^\pm$ are n-dimensional vectors whose elements are the incoming and outgoing pressure waves in the n bores. The physical constraints expressed in Eq. (3.59) are also generalized as

$$\begin{aligned} p_1 = p_2 = \ldots = p_N = p_J, \\ u_1 + u_2 + \ldots + u_N = 0. \end{aligned} \tag{3.65}$$

Calculations analogous to those outlined for the Kelly-Lochbaum junction yield

$$\boldsymbol{A} = \begin{bmatrix} \frac{2\Gamma_1}{\Gamma_J} - 1, & \frac{2\Gamma_2}{\Gamma_J}, & \cdots & \frac{2\Gamma_N}{\Gamma_J} \\ \frac{2\Gamma_1}{\Gamma_J}, & \frac{2\Gamma_2}{\Gamma_J} - 1, & \cdots & \frac{2\Gamma_N}{\Gamma_J} \\ \vdots & & \ddots & \vdots \\ \frac{2\Gamma_1}{\Gamma_J}, & \frac{2\Gamma_2}{\Gamma_J}, & \cdots & \frac{2\Gamma_N}{\Gamma_J} - 1 \end{bmatrix}, \qquad \text{where} \quad \Gamma_J = \sum_{l=1}^{N} \Gamma_l. \tag{3.66}$$

As an example, a 3-dimensional junction can be used to model an acoustic hole in a wind instrument: in this case, two waveguide sections represents the two sides of the acoustic bore with respect to the hole, and the third one represents the hole itself. Note also that when $N = 2$ Eq. (3.64) reduces to the Kelly-Lochbaum equations.

A second relevant extension of the Kelly-Lochbaum junction is the *loaded junction*, in which an external signal is injected into the system. A simple example is that of a string that is excited (e.g. hammered) at a given point. For continuity, the velocity of the string in this contact point will be the same at both sides. Moreover, during the contact this velocity will be equal to the velocity of the hammer. Finally, the sum of the forces at the contact point equals the hammer force. The following equations of continuity are then derived:

$$v_1 = v_2 = v_J, \qquad f_1 + f_2 + f_J = 0. \tag{3.67}$$

With the Kirchhoff analogies this is a series junction with an external load (the "currents" at the junction are the same, and the potentials at the junction sum to the driving potential). Then

$$\begin{aligned} f_J = &- f_1 - f_2 = \ldots = -2Z_0(v_1^+ + v_1^- - v_J), \\ \Rightarrow \quad & v_J = v_1^+ + v_2^+ - \frac{1}{2Z_0} f_J. \end{aligned} \tag{3.68}$$

This yields the scattering equations for the loaded junction:

$$\begin{aligned} v_1^-[n] =& v_J[n] - v_1^+[n] = v_2^+[n] + \frac{1}{2Z_0} f_J[n], \\ v_2^-[n] =& v_J[n] - v_2^+[n] = v_1^+[n] + \frac{1}{2Z_0} f_J[n]. \end{aligned} \tag{3.69}$$

The corresponding computational structure is shown in Fig. 3.14. This structure may be further extended to the case of N-dimensional parallel or series loaded junctions.

M-3.10

Implement the waveguide structure of Fig. 3.14. Add a loss filter (3.50) to each WG section.

3.4.3.3 Non-cylindrical geometries

A final remark is concerned with junctions of conical elements. Generalizing the cylindrical case is not straightforward, since the derivation of Kelly-Lochbaum equations is based on the implicit assumption of plane wave propagation. This assumption permits imposition of the constraints (3.59) on a flat scattering boundary, which is a wavefront for both p_1 and p_2. But wavefronts in conical sections are spherical and this circumstance makes it impossible to define a unique surface on which boundary conditions can be applied: Fig. 3.15(a) shows that there is a region between the two spherical wavefronts which is within neither conical segment. This ambiguity in the definition of the scattering boundary is usually overcome

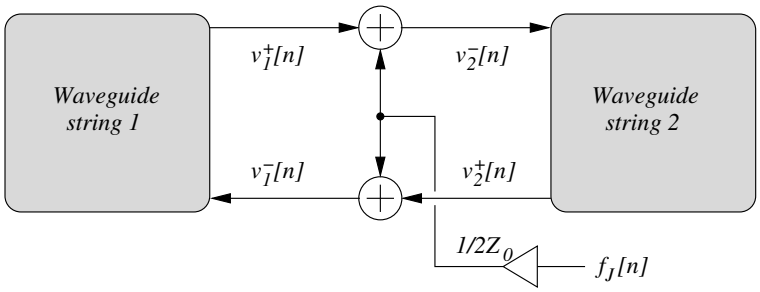


Figure 3.14: *Example of a loaded junction: a waveguide structure for a string excited by an external force signal $f_J[n]$ (e.g. a hammer).*

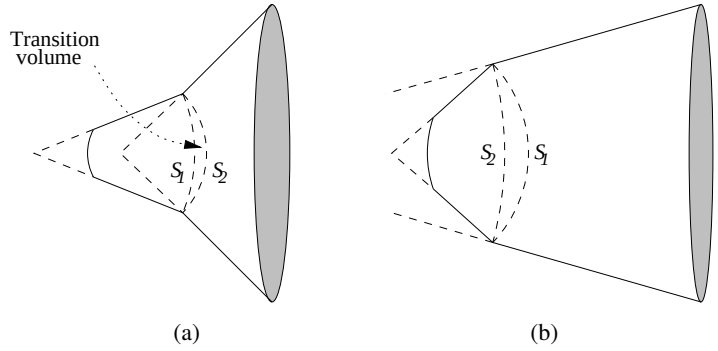


Figure 3.15: *Boundary regions for (a) non-convex and (b) convex conical junctions.*

by assuming that the transition volume is small and thus pressure is constant inside the volume. Under this assumption, continuity conditions analogous to (3.59) are imposed and the reflection coefficient ρ is generalized to a first order filter $R(s)$.

However, a second and more serious problem arises when one looks at the nature of $R(s)$. This filter turns out to be unstable (non-causal growing exponential) in the case of the convex configuration depicted in Fig. 3.15(b). While this circumstance is physically consistent (in the continuous-time domain the scattered waves can grow exponentially only for a limited time because they are cancelled out by subsequent multiple reflections), in a numerical simulation the system can turn out unstable, due to the approximations introduced by the discretization process and to round-off errors introduced by finite-precision.

3.5 Lumped models and the modal approach

Lumped modeling approaches can be applied in variety of contexts where the physical system under exam can be represented with ideal lumped elements. These include *electrical circuits*, with linear elements like capacities, resistances, and inductances, connected in series and parallel; *mechanical systems* viewed as ideal point masses connected through springs and dampers, representing mechanical resonators; *acoustic systems* viewed as networks of linear acoustic elements like bores, cavities, and acoustic holes (like the Helmoltz resonator examined previously).

Lumped models are particularly suited for describing systems whose spatial dimensions are small compared to acoustic wavelengths. As an example, pressure-controlled valves, such as single, double

or lip reeds, can be conveniently described using the lumped modeling paradigm. Although these systems are quite complicated, due to their limited spatial extensions they can be modeled using lumped elements, and it is widely accepted that such a simplified description captures the basic behavior of pressure controlled valves. Similar remarks hold for hammers and mallets: during collision, they are deformed and subject to internal losses and non-linear restoring forces. However, interactions with strings and bars have been modeled and efficiently implemented in sound synthesis algorithms by assuming the hammer/mallet to be a lumped mass.

3.5.1 Numerical methods

Unlike waveguide structures, lumped models are developed in the continuous-time domain, and are described through sets of ordinary differential equations (ODEs). In order to be implemented as numerical algorithms for sound synthesis, the differential equations have to be discretized in an efficient and effective manner. In most cases, a trade-off has to be found between accuracy of the discretization technique and efficiency of the resulting algorithms.

3.5.1.1 Impulse invariant method

When dealing with linear time-invariant systems, the most elementary technique to turn a continuous-time system into a discrete-time one is sampling its impulse response.

If a continuous-time LTI system is described in terms of Kirchhoff variables, then it is possible to define a transfer function which coincides with the admittance $\Gamma(s)$ of the system. As an example, in a mechanical lumped system this corresponds to defining the input as a driving force and the output as the resulting velocity. The inverse Laplace transform $\gamma(t)$ is the continuous-time impulse response. The linear system can thus be digitized by defining the discrete response as $\gamma_d[n] \triangleq \gamma(nT_s)$, i.e. by sampling $\gamma(t)$. This technique is widely used in the context of digital filter design, and it is usually termed the *Impulse invariant method*.

Assume that the continuous-time system has a rational transfer function $\Gamma(s)$. This can be rewritten using a partial fraction expansion (similarly to what we have done in Chapter *Fundamentals of digital audio processing* for discrete-time systems):

$$\Gamma(s) = \frac{B(s)}{A(s)} = \frac{\sum_{k=0}^{M} b_k s^{M-k}}{\sum_{k=0}^{N} a_k s^{N-k}}, \qquad \Rightarrow \quad \Gamma(s) = \sum_{k=1}^{N} \frac{K_k}{s - p_k}, \tag{3.70}$$

where the p_k's are the poles of the system. By taking the inverse Laplace transform of this latter equation, one can see that the impulse response $\gamma(t)$ is a combination of complex exponentials. This impulse response is then sampled to obtain its digital counterpart:

$$\gamma(t) = \sum_{k=1}^{N} K_k e^{p_k t}, \qquad \Rightarrow \quad \gamma_d[n] \triangleq \gamma(nT_s) = \sum_{k=1}^{N} K_k \left(e^{p_k T_s}\right)^n. \tag{3.71}$$

Finally, taking the Z-transform of γ_d yields:

$$\Gamma_d(z) = \sum_{k=1}^{N} \frac{K_k}{1 - p_{d,k} z^{-1}} = \frac{B_d(z)}{A_d(z)}, \qquad \text{with} \quad p_{d,k} = e^{p_k T_s}. \tag{3.72}$$

This equation tells that the transfer function $\Gamma_d(z)$ of the discretized system is still rational, with N poles $p_{d,k}$ uniquely determined by the continuous-time poles p_k.

One quality of the method is that stability is guaranteed at any sampling rate: if the continuous-time system is stable, i.e. $\mathrm{Re}(p_k) < 0$ for all k, then Eq. (3.72) tells that $|\, p_{d,k} \,| < 1$ for all k, i.e. the discrete time system is also stable. On the other hand, a drawback of the method is *aliasing*. Since $\gamma_d[n]$ has been obtained by sampling $\gamma(t)$, then the discrete-time response Γ_d is a periodization of Γ:

$$\Gamma_d(e^{j\omega}) = \sum_{k=-\infty}^{+\infty} \Gamma\left(\frac{j\omega}{T_s} + j\frac{2k\pi}{T_s}\right). \tag{3.73}$$

As a consequence, aliasing can occur in Γ_d if the bandwidth of Γ exceeds the Nyquist frequency.

3.5.1.2 Finite differences and mappings "s-to-z"

An alternative approach to the discretization of ODEs amounts to replacing time derivatives with *finite differences*, thus turning the differential equations directly into difference equations. Since in the Laplace domain the derivation operator is turned to a multiplication by s, and since in the z-domain the unit delay is turned into a multiplication by z^{-1}, approximating derivatives with finite differences corresponds to finding appropriate *s-to-z mappings*. Let $s = g(z)$ be such a mapping, then if the original continuous-time system is LTI with impulse response $\Gamma(s)$, the discrete-time response is found as $\Gamma_d(z) = \Gamma\left(g(z)\right)$.[6]

The simplest possible mapping is obtained by replacing the derivative with an incremental ratio. Let $x(t)$ be a generic smooth function of time, then

$$\frac{dx}{dt}(nT_s) = \lim_{h\to 0_+} \frac{x(nT_s) - x(nT_s - h)}{h} \approx \frac{x[n] - x[n-1]}{T_s} \quad \Rightarrow \quad s \approx \frac{1 - z^{-1}}{T_s} \triangleq g_1(z). \tag{3.74}$$

The mapping $g_1(z)$ is known in numerical analysis as the *backward Euler method*. The adjective "backward" is used because the first derivative of x at time n is estimated through the values of x at time n and $n-1$. Higher-order derivatives can be estimated through iterate application of Eq. (3.74). As an example, the second derivative is computed as

$$\frac{d^2x}{dt^2}(nT_s) \approx \frac{1}{T_s}\left[\frac{x[n] - x[n-1]}{T_s} - \frac{x[n-1] - x[n-2]}{T_s}\right] = \frac{x[n] - 2x[n-1] + x[n-2]}{T_s^2}. \tag{3.75}$$

Alternatively, a centered estimate is also often used in combination with the backward Euler method. In this case the second derivative is computed as:

$$\frac{d^2}{dt^2}x(t_n) \approx \frac{x[n+1] - 2x[n] + x[n-1]}{T_s^2}. \tag{3.76}$$

A second, widely used s-to-z mapping is provided by the *bilinear transform*. Like the backward Euler method, it can be seen as a finite approximation of the time derivative, but in this case the incremental ratio is assumed to approximate the value of $\dot{x}(t)$ averaged on time instants nT_s and $(n-1)T_s$:

$$\frac{\dot{x}(nT_s) + \dot{x}((n-1)T_s)}{2} \approx \frac{x[n] - x[n-1]}{T_s}, \qquad \Rightarrow \quad s \approx 2F_s\frac{1 - z^{-1}}{1 + z^{-1}} \triangleq g_2(z). \tag{3.77}$$

The mapping $g_2(z)$ is known in numerical analysis as the one-step *Adams-Moulton method*.

Both the backward Euler method and the bilinear transform are *implicit* numerical methods. This means that both methods turn a generic first-order differential equation $\dot{x}(t) = f(x(t), t)$ into a difference

[6] Note however that, unlike the impulse invariant method, finite differences do not assume linearity and time invariance of the original system, and are therefore more general methods.

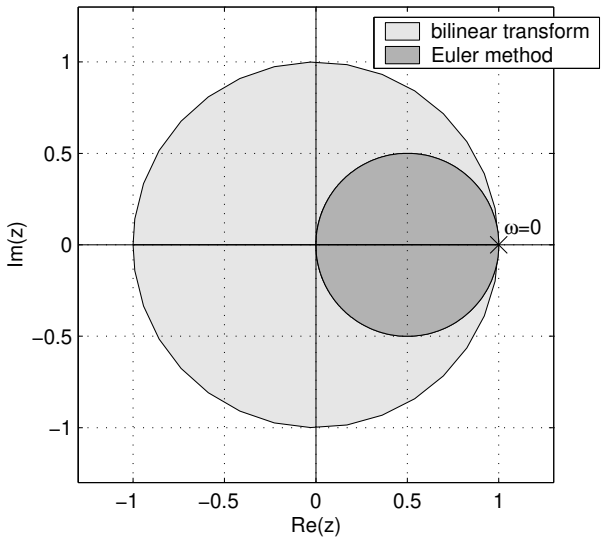


Figure 3.16: *Mapping of the vertical axis $s = j\omega$ (solid circle lines) and of the left-half s-plane (shaded regions) using the backward Euler method g_1 and the bilinear transform g_2.*

equation of the form $x[n] = f_d(x[n], x[n-1], n)$, in which $x[n]$ depends implicitly on itself through the function f_d. This is a source of problems for the resulting discrete-time system, since the difference equation is not computable explicitely due to the instantaneous dependence of a variable on itself. Below we discuss briefly this computability problem in the case of linear systems. Note that one advantage of the centered estimate (3.76) is that when it is applied in conjunction with the Euler method to a second-order ODE it leads to an *explicit* difference equation.

3.5.1.3 Accuracy, stability, computability

A comparison between the first estimate in Eq. (3.77) and the first in Eq. (3.74), gives the intuition that the bilinear transform provides a more accurate approximation than the Euler method. A rigorous analysis would show that the order of accuracy of the bilinear transform is two, while that of the backward Euler method is one.

Another way of comparing the two techniques consists in studying how the frequency axis $s = j\omega$ and the left-half plane $\mathrm{Im}(s) < 0$ are mapped by $g_{1,2}$ into the discrete domain. This provides information on both stability and accuracy properties of $g_{1,2}$. As shown in Fig. 3.16, both the methods define one-to-one mappings from $s = j\omega$, onto two circles. Therefore no frequency aliasing is introduced. Second, both the methods are stable, since the left-half s-plane is mapped inside the unit circle by both g_1 and g_2. However we also see that both mappings introduce *frequency warping*, i.e. the frequency axis is distorted. The bilinear transform g_2 maps the axis $s = j\omega$ exactly onto the unit circle $z = e^{j\omega_d}$, and the mapping between the continuous frequency ω and the digital frequency ω_d can be written analytically:

$$j\omega = \frac{2}{T_s}\frac{1 - e^{-j\omega_d}}{1 + e^{-j\omega_d}} = \frac{2j}{T_s}\tan\left(\frac{\omega_d}{2}\right), \qquad \Rightarrow \quad \omega_d = 2\arctan\left(\frac{\omega T_s}{2}\right). \tag{3.78}$$

At low frequencies ω_d increases almost linearly with ω, while higher frequencies are progressively compressed (warped) and $\omega_d \to \pm\pi$ as $\omega \to \pm\infty$. This warping phenomenon is the main drawback of the bilinear transform.

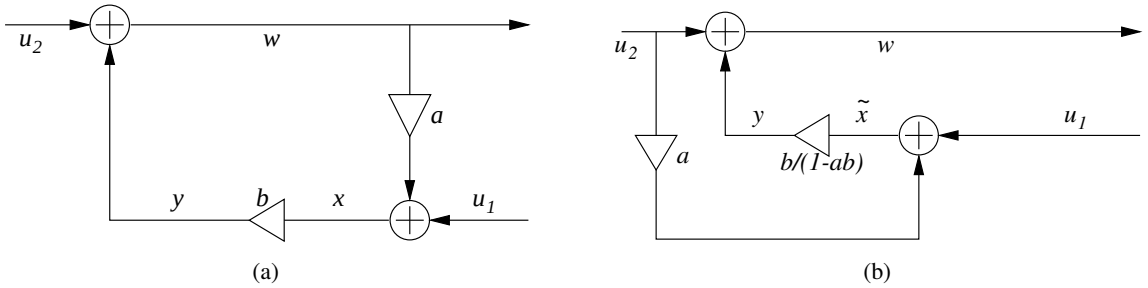


Figure 3.17: *A linear discrete-time system; (a) delay-free path, (b) equivalent realization with no delay-free paths.*

For the Euler method no analytic mapping can found from ω to ω_d. The function g_1 c "doubly" warps the frequency axis: there is a progressive warping in the direction of increasing frequency (similarly to the bilinear transform), and there is also warping normal to the frequency axis. Figure 3.16 also shows that the poles of the discrete-time system obtained with g_1 are more "squeezed" inside the unit circle than those obtained with g_2. Furthermore, it can happen that continuous-time poles with positive real-part are turned by g_1 into discrete-time poles with modulus less than unity: in other words g_1 can turn unstable continuous systems into stable discrete systems. This *numerical damping* is a second major drawback of the Euler method.

One more relevant aspect to discuss is the *computability* of the discrete-time systems obtained when discretizing a system of ODEs with either $g_{1,2}$ (or other mappings). As already stated, being these implicit methods the resulting difference equations are implicit. In order to clarify this point, let us consider the simple example depicted in Fig. 3.17(a). This system can be written as

$$\begin{cases} w[n] = \tilde{w}[n] + y[n], & \text{with} \quad \tilde{w} = u_2, \\ x[n] = \tilde{x}[n] + ay[n], & \text{with} \quad \tilde{x} = u_1 + au_2, \\ y[n] = bx[n], & \Rightarrow \quad y[n] = b[u_1[n] + au_2[n] + ay[n]], \end{cases} \tag{3.79}$$

where we have defined tilded variables $\tilde{w}$ and $\tilde{x}$ than only depend on the external inputs $u_{1,2}$, and are therefore known at each time n.

The signals y and x are connected through a *delay-free loop* and the resulting set of difference equations is implicit: in particular the last of Eqs. (3.79) shows that $y[n]$ depends implicitly on itself. It is easy, however, to rearrange the computation in order to solve this problem: the last of Eqs. (3.79) can be inverted, yielding

$$y[n] = \frac{b}{1-ab}[u_1[n] + au_2[n]]. \tag{3.80}$$

This new equation relates y to the computable vector $\tilde{x}$. Therefore, an equivalent realization of the system is obtained as shown in Fig. 3.17(b). The key point in this example is that the discrete-time system is linear, which allows explicit inversion of the last equation in (3.79).

This simple example is an instance of the so-called *delay-free loop* problem. In the linear case the literature of digital signal processing provides techniques for the restoring computability by rearrangement of the structure.

3.5.1.4 Wave digital filters

The bilinear transform finds application in the theory of Wave Digital Filters (*WDFs*). These structures are digital equivalents of the lumped circuit elements described in Sec. 3.2.1. WDF theory has been developed primarily for electric circuits but can be applied as well to mechanical and acoustic systems using Kirchhoff analogies.

Wave digital filters are constructed in two steps. The first step amounts to converting the continuous-time lumped circuits in wave variables. Here the definition of wave variables is identical to that used for waveguides models (see Eq. (3.43), namely:

$$f^+ = \frac{f + Z_0 v}{2}, \qquad f^- = \frac{f - Z_0 v}{2}, \tag{3.81}$$

where the mechanical Kirchhoff variables force f and velocity v have been used for clarity. The only and fundamental difference with Eq. (3.43) is that here Z_0 is a reference impedance that can be given any value and has no direct physical interpretation. The variables $f^\pm$ themselves do not have a clear physical interpretation since in a lumped model they cannot be interpreted as traveling waves. Therefore Eqs. (3.81) have to be regarded as a mere change of coordinates.

Using wave variables, circuit elements can be converted into *one-port elements*. Given one of the elementary lumped elements analyzed in Sec. 3.2.1 (resistance, inductance, capacity) and its associated impedance $Z(s)$, the new variables $f^\pm$ are related to each other through a *reflectance* $R(s)$:

$$F(s) = Z(s)V(s), \qquad \Rightarrow \quad F^-(s) = R(s)F^+(s), \quad \text{with} \quad R(s) \triangleq \frac{Z(s) - Z_0}{Z(s) + Z_0}. \tag{3.82}$$

The circuit element can then be visualized as a black box with a port consisting of two terminals, with a port voltage applied across them, and an associated flowing current, as in Fig. 3.18(a). A linear system can then be modeled through series and parallel connections of one-port elements: as an example, Fig. 3.18(b) visualizes a series connection of two ports representing the mechanichal system

$$m\ddot{x}(t) + kx(t) = f(t). \tag{3.83}$$

The second step in WDF design is the discretization of $R(s)$. The equivalent wave digital filter is obtained using the bilinear transform as $R(g_2(z))$. Note that since the reference impedance Z_0 can be given any value, this provides an additional degree of freedom in the design. In particular, Z_0 can be chosen such that the WDF has no delay-free paths from input to output, therefore guaranteeing computability when connecting more than one element. As an example, consider the three elementary mechanical impedances $Z_{\text{mass}}(s) = ms$, $Z_{\text{spring}}(s) = k/s$, $Z_{\text{loss}}(s) = r$. For the mass, the reflectance is $R_{\text{mass}}(s) = (ms - Z_0)/(ms + Z_0)$, therefore the equivalent WDF is

$$R_{\text{mass}}(z) = \frac{(2F_s - Z_0/m) - (Z_0/m + 2F_s)z^{-1}}{(2F_s + Z_0/m) - (Z_0/m - 2F_s)z^{-1}}, \quad \Rightarrow R_{\text{mass}}(z) = z^{-1} \text{ with } Z_0 = 2F_s m. \tag{3.84}$$

Therefore choosing $Z_0 = 2F_s m$ leads to the interesting result that no delay-free path is present in the corresponding WDF. Similarly, one can prove that $R_{\text{spring}}(z) = z^{-1}$ with $Z_0 = k/2F_s$, and $R_{\text{loss}} = 0$ with $Z_0 = r$.

This brief section has shown that WDFs can be used to digitize lumped element networks using wave variables and adapted impedances in such a way that delay-free computational loops are avoided in the resulting numerical structure. We have shown a single example of a series connection between two elements. The concept of connection is generalized in WDF theory with the concept of *adaptors*, which are N-port elements that model interconnection between arbitrary numbers of elements.

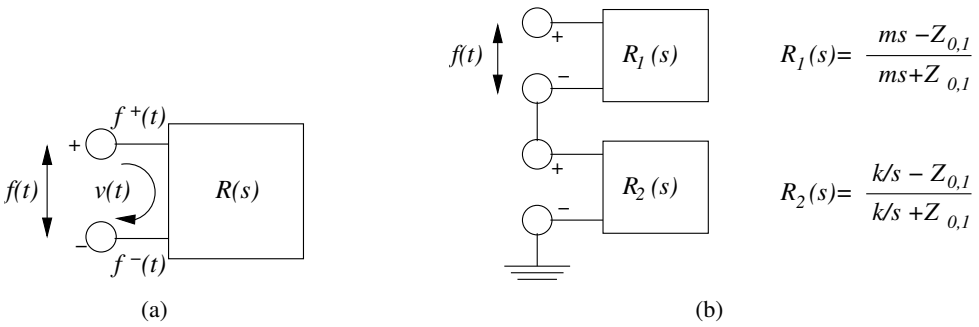


Figure 3.18:

3.5.2 Modal synthesis

Modal synthesis is conceptually simple: the sound of a resonating object is represented as a linear combination of the outputs of N second order oscillators, each of which represents one mode of oscillation of the object excited by a driving force or acoustic pressure: in this sense modal synthesis can be regarded as a lumped physical modeling approach, and can also be interpreted a source-filter approach in which the source is the driving signal and the filter is bank of second-order resonators.

Understanding the mathematical and physical basis of modal theory is a bit less straightforward. In the next sections we sketch the main concepts for both discrete systems (e.g. discrete networks of masses, springs, and dampers) and continuous systems (i.e. partial differential equations in space and time). We show that modal theory is fundamentally the same for these classes of systems.

Therefore the power of modal synthesis is that it is a very general technique that can be applied to a large class of sounding physical systems (while e.g. waveguide techniques are suited only for elastic systems that obey some perturbed version of the D'Alembert equation).

3.5.2.1 Normal modes in finite dimensional systems

In Sec. 3.2.1 we have studied the simple example of two coupled mechanical oscillators, and we have seen that the resulting system can be viewed as the combination of two *uncoupled* oscillators, whose frequencies depend on those of the original ones. This approach can be extended to a generic network of N linear undamped oscillators:

$$\boldsymbol{M}\ddot{\boldsymbol{y}}(t) + \boldsymbol{K}\boldsymbol{y}(t) = \boldsymbol{f}_{\text{ext}}(t). \tag{3.85}$$

In this equation $\boldsymbol{y}$ is a vector containing the displacements of the N points of the network, while $\boldsymbol{M}$ is the mass matrix: typically (but not necessarily) it is diagonal and contains the masses m_l ($l = 1 \dots N$)of each point of the network. $\boldsymbol{K}$ is the stiffness matrix and is in general not diagonal because the points are coupled through springs.

Now we consider the homogeneous equation ($\boldsymbol{f}_{\text{ext}} \equiv 0$) and look for a factorized solution of the form $\boldsymbol{y}(t) = \boldsymbol{s} \cdot \sin(\omega t + \phi)$. By substituting this into Eq. (3.85), one finds

$$\boldsymbol{K}\boldsymbol{s} = \omega^2 \boldsymbol{M}\boldsymbol{s}. \tag{3.86}$$

This is a generalized eigenvalue problem for the matrix $\boldsymbol{K}$: more precisely, ω^2 is an eigenvalue of $\boldsymbol{M}^{-1}\boldsymbol{K}$ and $\boldsymbol{s}$ is the associated eigenvector. In general one will find N distinct eigenvalues and eigenvectors ω_i^2 and $\boldsymbol{s}_i$ (for simplicity we consider normalized $\boldsymbol{s}_i$'s). The key property of these eigenvectors

is that they are orthogonal with respect to the mass and the stiffness matrix:

$$\boldsymbol{s}_j^T \boldsymbol{M} \boldsymbol{s}_i = \delta_{i,j} m_i, \qquad \boldsymbol{s}_j^T \boldsymbol{K} \boldsymbol{s}_i = \delta_{i,j} k_i, \tag{3.87}$$

where m_i and $k_i = \omega_i^2 m_i$ are real positive scalars. The orthogonality condition also implies that the *modal shapes* $\boldsymbol{s}_i$ are linearly independent. The $\boldsymbol{s}_i$'s can be used to define a *modal transformation*, i.e. a change of spatial coordinates that transforms system (3.85) into a set of N uncoupled oscillators:

$$\boldsymbol{y} = \boldsymbol{S}\boldsymbol{q} \quad \boldsymbol{q} = \boldsymbol{S}^T \boldsymbol{y}, \qquad \text{with} \quad \boldsymbol{S} = [\boldsymbol{s}_1 | \boldsymbol{s}_2 | \dots | \boldsymbol{s}_N]\,. \tag{3.88}$$

Substituting this into Eq. (3.85) and premultiplying by $\boldsymbol{S}^T$ yields

$$\boldsymbol{M_q}\ddot{\boldsymbol{q}} + \boldsymbol{K_q}\boldsymbol{q} = \boldsymbol{S}^T \boldsymbol{f}_{\text{ext}}(t), \qquad \text{with} \quad \boldsymbol{M_q} = \boldsymbol{S}^T \boldsymbol{M} \boldsymbol{S}, \quad \boldsymbol{K_q} = \boldsymbol{S}^T \boldsymbol{K} \boldsymbol{S}. \tag{3.89}$$

By virtue of the orthogonality property, the matrices $\boldsymbol{M_q}$ and $\boldsymbol{K_q}$ are diagonal and contain the elements m_i and k_i on their diagonals, respectively. Therefore this is a system of uncoupled oscillators with frequencies ω_i, the quantities m_i and k_i represent the masses and the stiffnesses of these modes.

The matrix $\boldsymbol{S}^T$ of the modal shapes defines how a driving force $\boldsymbol{f}_{\text{ext}}$ acts on the modes: as a particular case, consider a scalar force acting only on the lth point of the network, i.e. $\boldsymbol{f}_{\text{ext}}(t) = [0, \dots, f_{\text{ext}}(t), \dots, 0]^T$ (where the only non-null element is in the lth index). This force is applied to the generic ith mode, scaled by the factor $\boldsymbol{s}_{i,l}$, i.e. the shape of the ith mode at the lth point of the network. If this factor is 0, i.e. if the ith mode has a *node* at the lth point of the network, then no force is transmitted to the mode.

The oscillation $y_l(t)$ of the system at the lth spatial point will be the sum of the modal oscillation weighted by the modal shapes, according to Eq. (3.88): $y_l(t) = \sum_{i=1}^{N} \boldsymbol{s}_{i,l} q_i(t)$. Again, if the ith mode has a *node* at the lth point of the network, that mode will not be "heard" in this point. In conclusion the motion of the network is determined by the motion of N second-order mechanical oscillators and by the transformation matrix $\boldsymbol{S}$.

This formalism can be extended to systems that include damping, i.e. where we add a term $\boldsymbol{R}\dot{\boldsymbol{y}}$ in Eq. (3.85).

3.5.2.2 Normal modes in PDEs

Now look at the concept of normal modes from a different perspective: a distributed object is not modeled as a network of lumped elements, but instead as a partial differential equation that describes the displacement $y(x,t)$ as a continuous function of space and time. We can reformulate the modal description even in this case: as we will see, there are strict analogies with the case of finite dimensional systems outlined above.

We use a concrete example, a string with fixed ends, to derive the modal formulation in the case of continuous systems. In analogy with the case of finite dimensional systems, we now state that a normal mode is a factorized solution $y(x,t) = s(x)q(t)$. For the example under exam, we already know that the D'Alembert equation with fixed boundary conditions admits the factorized solutions $y_n(x,t) = s_n(x)q_n(t)$ of the form (3.21). If a force density $f_{\text{ext}}(x,t)$ is acting on the string, the equation is

$$\mu \frac{\partial^2 y}{\partial t^2}(x,t) - T \frac{\partial^2 y}{\partial x^2}(x,t) = f_{\text{ext}}(x,t). \tag{3.90}$$

If one substitutes in this equation the mode $y_n(x,t)$, and then multiplies by $s_n(x)$ and integrates over the string length, the following equation is found:

$$\left[\mu \int_0^L s_n^2(x)dx\right] \ddot{q}_n(t) - \left[T \int_0^L s_n''(x) s_n(x) dx\right] q_n(t) = \int_0^L s_n(x) f_{\text{ext}}(x,t) dx. \tag{3.91}$$

The second integral can be integrated by parts to obtain

$$\left[\mu \int_0^L s_n^2(x)dx\right] \ddot{q}_n(t) - T\left[s_n'(x)s_n(x)\big|_0^L - \int_0^L [s_n'(x)]^2 dx\right] q_n(t) = \int_0^L s_n(x) f_{\text{ext}}(x,t)dx, \quad (3.92)$$

where the term $s_n'(x)s_n(x)|_0^L$ is identically zero for fixed (or even free) boundary conditions. Therefore the equation for the nth mode is that of a second-order oscillator with mass $m_n = \mu \int_0^L s_n^2(x)dx$ and stiffness $k_n = T\int_0^L [s_n'(x)]^2 dx$. For the ideal string the modal shapes are simply $s_n(x) = \sin(n\pi x/L)$, therefore $m_n = \mu L/2$ and $k_n = TL/2$.

The shape also defines how a driving force acts on the mode. As a particular case, consider a force density that is ideally concentrated in a single point x_{in} of the string, i.e. $f_{\text{ext}}(x,t) = \delta_D(x - x_{in})u(t)$ (where the function $\delta_D(\cdot)$ is the Dirac delta): then the force acting on the nth mode is $s_n(x_{in})u(t)$, and if x_{in} is a node of the mode then no force is transmitted to it. We already know that the oscillation $y(x_{out}, t)$ of the system at the spatial point x_{out} will be the sum of the modal oscillations weighted by the modal shapes: $y(x_{out}, t) = \sum_{n=1}^{+\infty} \boldsymbol{s}_n(x_{out})q_n(t)$. Again, if the nth mode has a *node* at the point x_{out}, that mode will not be "heard" in this point.

This analysis can be extended to include dispersion and dissipation. As an example, we know that for a dissipative string we have to add the terms $d_1\partial y/\partial t - d_2\partial/\partial t(\partial^2 y/\partial x^2)$ on the left-hand side of Eq. (3.90). Again, by substituting the mode $y_n(x,t)$ in the equation, and then multiplying by $s_n(x)$ and integrating over the string length, one finds that the term $\left[d_1 \int_0^L s_n^2(x)dx + d_2 \int_0^L [s_n'(x)]^2 dx\right] \dot{q}(t)$ has to be added to Eq. (3.92), which represents a viscous damping term for the second order oscillator.

M-3.11

Compute modal parameters for a string with linear dissipation.

M-3.11 Solution

```
function [omega,alpha,m,s]=modal_string(L,T,mu,d1,d2,N,M);

xstep=L/(M-1); xpoints=0:xstep:L;

s=zeros(N,M); omega=zeros(1,N); alpha=omega; m=omega;
for i=1:N % i is mode number
    s(i,:)= sin(i*pi*xpoints/L);    %spatial shape
    m(i) = mu*xstep*sum(s(i,:).^2); %=mu*L/2; modal mass
    dsdx= i*pi/L*cos(i*pi*xpoints/L);
    k = T*xstep*sum(dsdx.^2);       %=T*L/2*(i*pi/L)^2; modal stiffness
    omega(i)=sqrt(k/m(i));          %=i*pi*c/L; modal frequency
    alpha(i)=(d1*xstep*sum(s(i,:).^2)-d2*xstep*sum(dsdx.^2) )/(2*m(i)); %loss
end
```

Parameters are functions of the string tension T, linear density μ, loss factors $d_{1,2}$. One can choose the number N of modes to compute, and the number M of spatial points for the shape computation. Two remarks. First, we are using the ideal spatial shapes: this is not correct for the dissipative string, but is acceptable for small $d_{1,2}$ values. Second, we are computing the integrals numerically, although for the ideal string shapes these have analytical solutions: in Sec. 3.5.3 we will examine less trivial shapes.

In conclusion the modal representation of continuous systems described by PDEs is in strict analogy with that of discrete systems described as networks of masses and springs. Here we have obtained similar equations, where the discrete spatial index $l = 1 \ldots N$ indicating the points of the network (3.85) has become a continuous spatial variable x, sums over l have become integrals over x, and a numerable

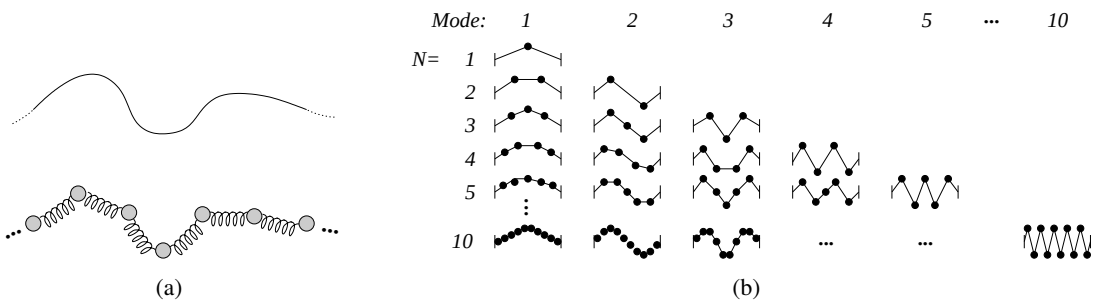


Figure 3.19: *Analogies between continuous and discrete systems: (a) approximation of an ideal string with a mass-spring network; (b) modes of the discrete system for different numbers N of masses.*

infinity of modes has been found instead of a finite set of N modes. These strict analogies reflect the fact that continuous systems can be seen as the limit of discrete systems when the number of masses becomes infinite. As an example, a string can be approximated with the discrete network of Fig. 3.19(a) made of N masses and $N+1$ springs. Figure 3.19(b) shows that for a given N the system has N modes, whose shapes resemble closely those of the first N modes of the continuous string. Moreover the approximation grows closer and closer as N increases. One could also show that the modal frequencies of the continuous system are underestimated by those of the discrete system, due to the spatial discretization.

3.5.2.3 Discrete-time mechanical oscillators

We have seen that each mode of either a discrete or a continuous system is a second order oscillator:

$$\begin{aligned} \ddot{q}(t) + 2\alpha\dot{q}(t) + \omega_0^2 q(t) &= \frac{1}{m} f_{\text{mode}}(t), \\ Q(s) = H(s)F_{\text{mode}}(s), \qquad &\text{with} \quad H(s) = \frac{m^{-1}}{s^2 + 2\alpha s + \omega_0^2}. \end{aligned} \tag{3.93}$$

The frequency $\omega_0 = k/m$ and the loss factor $\alpha = r/m$ depend on the geometry and the material of the object. The force f_{mode} that is "felt" by a single mode depends on the modal shape and on the spatial force distribution, and is scaled by the modal mass m. The displacement $y(x,t)$ at a certain point x of the structure is a linear combination of the modes $q(t)$, where the coefficients of the linear combination are the modal shapes $s(x)$ at the point x. This is true whether we have a discrete set of points x_i or a continuous domain, although in practice the spatial domain will be always discretized.

In order to construct a modal synthesizer, the first step to perform is to construct a discrete-time equivalent of the second order oscillator (3.93). We can discretize the differential equation with the numerical methods examined previously in Sec. 3.5.1. The impulse invariant method yields:

$$H(z) = \frac{\left[T_s \left(\frac{e^{-\alpha T_s}}{m\omega_r}\right) \sin(\omega_r T_s)\right] z^{-1}}{1 - \left[2e^{-\alpha T_s}\cos(\omega_r T_s)\right] z^{-1} + e^{-2\alpha T_s} z^{-2}}. \tag{3.94}$$

The presence of a z^{-1} factor at the numerator indicates that this is an explicit numerical scheme (there is no instantaneous dependence on the input). The backward Euler method yields

$$H(z) = \frac{\frac{1}{m(F_s^2 + 2\alpha F_s + \omega_0^2)}}{1 - \frac{2F_s(\alpha + F_s)}{F_s^2 + 2\alpha F_s + \omega_0^2} z^{-1} + \frac{F_s^2}{F_s^2 + 2\alpha F_s + \omega_0^2} z^{-2}}. \tag{3.95}$$

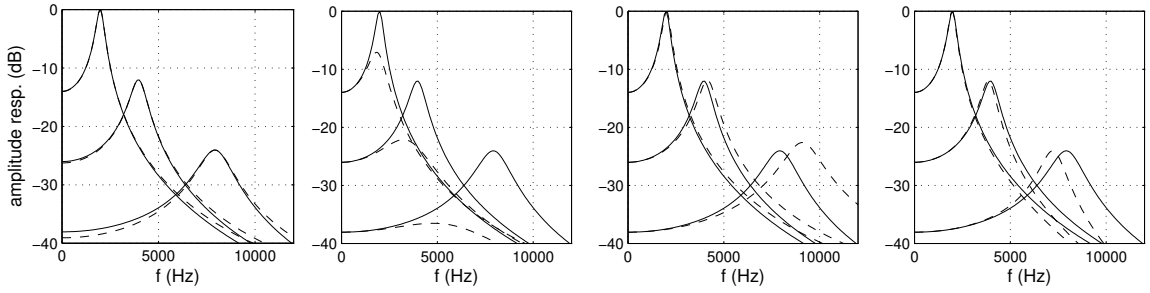


Figure 3.20: *Amplitude responses of a second order oscillator with constant mass and quality factor, and $\omega_0 = 2, 4, 8$ kHz: continuous-time responses (solid lines) and discrete-time responses (dashed lines) with (a) impulse invariant method, (b) backward Euler method, (c) backward Euler method with centered scheme, (d) bilinear transform.*

This instead is an implicit numerical scheme. The backward Euler method with centered scheme yields

$$H(z) = \frac{\frac{T_s^2}{m} z^{-1}}{1 + \left[\omega_0^2 T_s^2 + 2\alpha T_s - 2\right] z^{-1} + \left[1 - 2\alpha T_s\right] z^{-2}}. \tag{3.96}$$

Like in the impulse invariant case, this is an explicit numerical scheme. By looking at the poles of this discrete-time system one can see that it can become unstable depending on the mechanical parameters and on the sampling period: the scheme is not *unconditionally stable*. Finally, the bilinear transform yields

$$H(z) = \frac{\left[\frac{1}{m(4F_s^2 + 4\alpha F_s + \omega_0^2)}\right] (1 + 2z^{-1} + z^{-2})}{1 + \frac{2(\omega_0^2 - 4F_s^2)}{4F_s^2 + 4\alpha F_s + \omega_0^2} z^{-1} + \frac{4F_s^2 - 4\alpha F_s + \omega_0^2}{4F_s^2 + 4\alpha F_s + \omega_0^2} z^{-2}}. \tag{3.97}$$

Like in the case of the backward Euler method, this is an implicit numerical scheme.

The resulting amplitude responses are shown in Fig. 3.20. As expected, the impulse invariant method exhibits aliasing, the Euler method exhibits warping and numerical damping, the Euler method with centered scheme tends to become unstable for high $omega_0$ values, and the bilinear transform exhibits warping (but not numerical damping).

M-3.12

Write a function that computes the filter coefficients of the mechanical oscillator discretized with (a) the impulse invariant method, (b) the Euler method $g_1(z)$, (c) Euler method with the centered estimate (3.76), and (d) the bilinear transform. Compare the frequency responses of the resulting discrete-time systems.

M-3.12 Solution

```
function [B,A]=modal_oscillator(m,alpha,omega,method)

global Fs;  Ts=1/Fs;

switch method
case 'impinv'
    omegar=sqrt(omega^2-alpha^2);    eaTs=exp(-alpha*Ts);
    B= [0 Ts*eaTs/(m*omegar)*sin(omegar*Ts) 0];
```

```
    A= [1 -2*eaTs*cos(omegar*Ts) eaTs^2];
case 'euler'
    delta= Fs^2+2*alpha*Fs+omega^2;
    B= [1/(m*delta) 0 0];
    A= [1 -2*Fs*(alpha+Fs)/delta Fs^2/delta];
case 'eulercenter'
    B= [0 Ts^2/m 0];
    A= [1 (omega^2*Ts^2 +2*alpha*Ts -2) (1-2*alpha*Ts)];
case 'bilin'
    delta=4*Fs^2 +4*alpha*Fs +omega^2;
    B= [1/(m*delta) 2/(m*delta) 1/(m*delta)];
    A= [1 2*(omega^2 -4*Fs^2)/delta (4*Fs^2 -4*alpha*Fs +omega^2)/delta];
otherwise error('unknown numerical method');
end
```

3.5.2.4 A modal synthesizer

A simple modal synthesizer can be constructed as a parallel connection of N numerical oscillators. By choosing a different center frequency ω_0 and damping factor α for each oscillator, it is possible to account for a set of partials and decay times of the resonator spectrum. Moreover, the modal shapes determine both how a force signal is injected into the modal oscillator and how the modal oscillations are combined

M-3.13

Write a function that computes the output of a modal resonator given an input force signal.

M-3.13 Solution

```
function y = modal_synth(x,omega,alpha,m,s,in,out,method);

global Fs;
N=length(omega);% it must be size(omega)=size(alpha)=size(m)=N
                % it must be size(s,1)=N; 0<in<size(s,2); 0<out<size(s,2);
y=zeros(1,length(x));
for i= 1:N
    [B,A]=modal_oscillator(m(i),alpha(i),omega(i),method);
    y_i=filter(B,A,s(i,in)*x);
    y=y +s(i,out)*y_i;
end
```

We have assumed that the force distribution is concentrated in a single point, represented by the index `in`. We "pick-up" the resonator signal at another point, represented by the index `out` (like we were using a contact mike attached to the object at the point `out`).

The input modal parameters can be chosen to match those of an arbitrary object. Moreover, morphing between different shapes and material can be obtained by designing appropriate trajectories for these parameters.

M-3.14

Synthesize the sound of a dissipative string using the modal approach.

M-3.14 Solution

```
global Fs; Fs=44100;
slength=8; %sound length (s)

%%%%% Physical parameters for a E3 guitar nylon string %%%%%
mu=5.25e-3;
T = 60; % string tension (N)
L= 0.65; %string length (m)
d1=mu*.65;
d2=-T*9e-8;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fmax=30; % impulsive force (N)
x=[fmax zeros(1,round(slength*Fs))]; %force signal zero-padded to sound length

N= 120; M=round(L/5e-3);        %no. of modes and spatial points (incl.ends)
in=round(M/20); out=round(M/20); %input and output points
[omega,alpha,m,s]=modal_string(L,T,mu,d1,d2,N,M); %compute modal parameters
y=modal_synth(x,omega,alpha,m,s,in,out,'impinv'); %compute output signal
```

3.5.3 Modal analysis

The modal synthesizer that we have constructed needs to know the modal parameters for the specific resonator under exam. The question is then how to determine these parameters.

In the case of a discrete system of N point masses with linear interaction forces, modal parameters are exactly found through standard matrix calculations. Most systems of interest of course do not fit these assumptions. For some distributed systems, particularly for symmetrical problems with simple boundary conditions, the partial differential equation describing the system can be solved analytically, giving the modal parameters. Alternatively, either accurate numerical simulations (e.g. wave-guide mesh methods) or "real" physical measurements can be used.

3.5.3.1 Simple 1-D shapes

The simplest tractable case is the ideal string: we have already discussed the modal solution in this case. There are other tractable cases: one interesting example is the ideal bar, with various boundary conditions. Bars are almost as relevant as strings for musical applications: mallet percussion instruments, such as the marimba, the xylophone, the vibraphone, and so on, are based on the oscillations of bars.

Transverse vibrations in a bar are due to internal elastic force generated when the bar is bent. One can show that for a bar with constant cross-section the equation governing the bar transversal displacement y is the Euler-Bernoulli equation:

$$\frac{\partial^2 y}{\partial t^2}(x,t) = -\frac{EK^2}{\rho}\frac{\partial^4 y}{\partial x^4}(x,t), \tag{3.98}$$

where E is the Young modulus of the material, K is the radius of gyration,[7] and ρ is the volume density. Note that the fourth-order term is the one that we used to describe a stiff (and dispersive) string. The modal solutions $y(x,t) = s(x)q(t)$ are in this case

$$y(x,t) = [A\cosh kx + B\sinh kx + C\cos kx + D\sin kx]\cos(\omega t + \phi), \quad \text{with} \quad k = c\omega, \tag{3.99}$$

[7] This would need some explanation. In short: $K^2 = \frac{1}{S}\int z^2 dS$, where $S = \int dS$ is the total cross-section of the bar and z is the distance from the neutral axis, i.e. the axis along the bar which does not change its length when the bar is bent (at one side of the neutral axis there is elongation, at the other side there is compression). Everything clear??

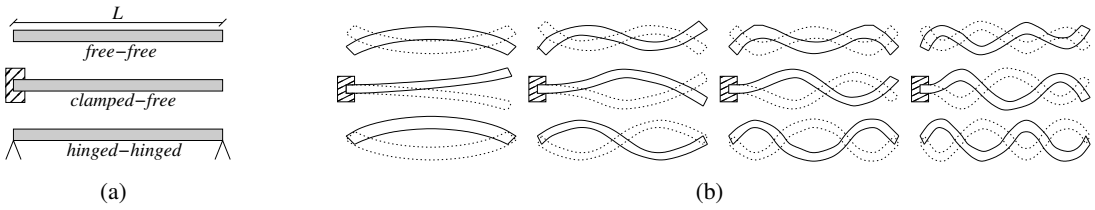


Figure 3.21: *Modal description of the ideal bar: (a) ideal bar with various boundary conditions and (b) corresponding modes.*

and where $c^2 = \omega K\sqrt{E/\rho}$. This modal solution cannot be interpreted in terms of traveling waves, therefore waveguide methods fall short here, while modal synthesis can be successfully employed.

The constants A, B, C, D as well as the allowed frequencies are determined depending on four boundary conditions (two at each end). The conditions for a *free* end are $\partial^2 y/\partial x^2 = \partial^3 y/\partial x^3 = 0$ (no torque and no shearing force); those for a *supported* (hinged) end are $y = \partial^2 y/\partial x^2 = 0$ (no displacement and no torque); and those for a *clamped* end are $y = \partial y/\partial x = 0$ (no displacement and zero slope). Three notable examples are shown in Fig. 3.21(a). For these cases, numerical solution of the equations resulting from boundary conditions yields

$$\begin{aligned}
\text{(free-free)} \quad \{\omega_n\} &= \frac{\pi^2 K}{4L^2}\sqrt{\frac{E}{\rho}}\left[3.011^2, 5^2, 7^2, \ldots, (2n+1)^2, \ldots\right], \\
\text{(clamped-free)} \quad \{\omega_n\} &= \frac{\pi^2 K}{4L^2}\sqrt{\frac{E}{\rho}}\left[1.194^2, 2.988^2, 5^2, \ldots, (2n-1)^2, \ldots\right], \\
\text{(hinged-hinged)} \quad \{\omega_n\} &= \frac{2\pi^2 K}{L^2}\sqrt{\frac{E}{\rho}}n^2.
\end{aligned} \tag{3.100}$$

Note that in the first two cases the frequencies are strongly inharmonic, while in the third case they are harmonically related: the corresponding lowest modes are shown in Fig. 3.21(b). Mallet percussions most typically use bars with (approximately) free-free conditions. However in many cases their bars do not have constant cross-sections, instead their are cut with an arch on the underside in such a way that the theoretical partials of the free-free series in Eq. (3.100) are shifted and aligned to an almost harmonic series.

M-3.15

Compute modal parameters for a bar with the three boundary conditions examined here, and with linear dissipation.

M-3.15 Solution

Like Example M-3.11, but using the modal shapes of the ideal bar.

3.5.3.2 Simple 2-D shapes

The first example of a musically relevant 2-D shape is a rectangular membrane with fixed ends, like the one depicted in Fig. 3.22(a). The ideal membrane obeys the 2-D D'Alembert equation:

$$\sigma\frac{\partial^2 z}{\partial t^2}(x,y,t) = T\left[\frac{\partial^2 z}{\partial x^2}(x,y,t) + \frac{\partial^2 z}{\partial y^2}(x,y,t)\right] = T\nabla^2 z(x,y,t), \tag{3.101}$$

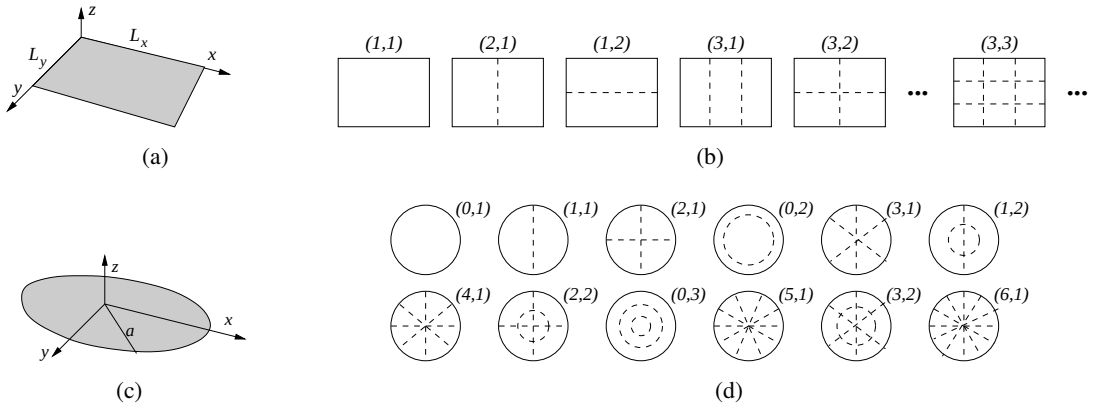


Figure 3.22: *Modal description of ideal membranes: (a) ideal rectangular membrane with fixed ends and (b) corresponding modes; (c) ideal circular membrane with fixed ends and (d) corresponding modes.*

where z is the membrane vertical displacement and the constants T and σ are the membrane surface tension (in N/m) and surface density (in Kg/m^2). The symbol $\nabla^2 = \partial^2/\partial x^2 + \partial^2/\partial y^2$ stands here for the 2-dimensional Laplacian operator. Modal solutions $z(x, y, t) = s^{(x)}(x)s^{(y)}(y)q(t)$ are found with the same procedure used for the ideal string:

$$\begin{aligned} z_{n,m}(x, y, t) &= \sqrt{\frac{2}{L_x}}\sqrt{\frac{2}{L_y}} \sin\left(k_n^{(x)} x\right) \sin\left(k_m^{(y)} y\right) \cos\left(\omega_{n,m} t + \phi_{n,m}\right), \\ \text{with} \quad k_n^{(x)} &= \frac{n\pi}{L_x}, \quad k_m^{(y)} = \frac{m\pi}{L_y}, \quad \omega_{n,m} = c\sqrt{\left[k_n^{(x)}\right]^2 + \left[k_n^{(x)}\right]^2}. \end{aligned} \tag{3.102}$$

Note that the modal frequencies $\omega_{n,m}$ are not harmonically related in this case. The modal shapes $s_{n,m}(x, y) = s_n^{(x)}(x)s_m^{(y)}(y)$ have straight nodal lines: the lowest modes are shown in Fig. 3.22(b).

A second example, even more relevant for musical applications, is the circular membrane with fixed ends, like the one in Fig. 3.22(c). In this case the 2-D D'Alembert equation is more conveniently written in circular coordinates $x = r\sin\theta$ and $y = r\cos\theta$ and the laplacian becomes $\nabla^2 = \partial^2/\partial r^2 + 1/r(\partial/\partial r) + 1/r^2(\partial/\partial\theta)$. Accordingly, one looks for modal solutions of the form $z(r, \theta, t) = s^{(r)}(r)s^{(\theta)}(\theta)q(t)$.

Substituting this into the 2-D D'Alembert equation results in two differential equations for $s^{(r)}$ and $s^{(\theta)}$. One finds the angular shapes $s_m^{(\theta)}(\theta) = \cos(m\theta)$. Then for each m, the radial shapes are $s_n^{(r)}(r) = J_m\left(k_{m,n}^{(r)} r\right)$, i.e. they are the first-kind Bessel functions of order m, with radial frequencies $k_{m,n}^{(r)}$. The allowed values for $k_{m,n}^{(r)}$ are found as usual by imposing that $s_n^{(r)} = 0$ at the fixed boundary, therefore are determined by the nth zero of J_m. In conclusion the m, n mode has m nodal diameters (determined by the function $s_m^{(\theta)}$) and n nodal circles (determined by the function $s_n^{(r)}$). The lowest modal frequencies $\omega_{n,m}$ are

$$\{\omega_{n,m}\} = \frac{2.405c}{a}\left[1, 1.594, 2.136, 2.296, 2.653, 2.918, 3.156, 3.501, 3.6, 3.652, 4.06, 4.154\right], \tag{3.103}$$

and are highly inharmonic. The corresponding modes are shown in Fig. 3.22(d).

M-3.16

Compute modal parameters for a rectangular and a circular bar with fixed boundary conditions, and with linear dissipation.

M-3.16 Solution

Like Example M-3.11, but using the modal shapes of the ideal rectangular and circular membrane.

3.5.3.3 Experimental estimation

When the modal solution cannot be written analytically, modal parameters can still be estimated. One approach is to extract modal data from a recorded audio signal. Various methods are known that can estimate resonances (center frequencies and quality factors) from a signal: these include linear prediction techniques and partial tracking techniques examined in Chapter *Sound modeling: signal based approaches*. However there are various problems to deal with. First, modal frequencies are often very closely spaced and one needs high-resolution methods that are able to discriminate nearby resonances. Second, estimates derived from analysis of a single sound lack information about the spatial shapes of the modes. Third, there are many inaccuracies related to technical difficulties in the recording of object responses: ideally one should record the impulse response of an object, for many different excitation points and many different pick-up points. In practice one will strike the object and record the response in air, with consequent spatially distributed interactions, and sound radiation through air.

Modal shapes may be observed through more sophisticated measurement devices, e.g. by using holographic interferometry. A relatively simple experimental technique amounts to place some finely divided material (e.g. fine sand or flour) on the resonating object (e.g. a plate of arbitrary shape), and then setting the object into forced oscillation (most typically through mechanical or electromechanical means) with a sinusoidal driving signal which is tuned to the frequency of the desired mode. As a consequence one will observe the sand on the object bouncing and moving about, and only at or near the nodal lines of the mode the sand will be stationary. Thus the sand is either bounced off the object or else collects at the nodes, forming so-called *Chladni patterns* (from the name of the german physicist and musician who first observed nodal patterns through this technique). Variations of this technique have been commonly used by acoustic instrument makers, especially for the design and construction of the resonating bodies of violins, guitars, cellos, etc.

An alternative "experimental" approach amounts to simulate the response of an object with finite difference or finite element methods. This implies spatial discretization, which means that only a finite amount of modes can be estimated. Moreover, modal data obtained in this way suffers from underestimation of modal frequencies, due to errors introduced by spatial discretization.

3.6 Non-linear physical models

So far in this Chapter we have examined linear models, mostly employed to simulate physical resonators. However musical oscillators are often strongly non-linear.

Non-linearities must be present for a system to reach stable self-sustained oscillations. As an example, self-sustained oscillations in the acoustic bore of a woodwind or brass instrument can only be explained in terms of a non-linear, persistent excitation mechanism. More precisely, the valve (a single or double-reed, or the player's lips) at the bore termination acts as a non-linear element that injects energy into the system. A very similar description holds for bowed string instruments, where the bow and its non-linear friction force is the exciting element. In other cases the instrument is non-linearly excited only for a limited amount of time: a struck string or bar interacts with the hammer or mallet through a non-linear contact force. Values for the contact time are typically a few milliseconds, and after this short excitation the system evolution is linear and the oscillations decay away.

Generalizing from the above examples, we may schematize a musical instrument (or any sound-producing physical system) by means of two main functional blocks, as in Fig. 3.23. The *resonator* is

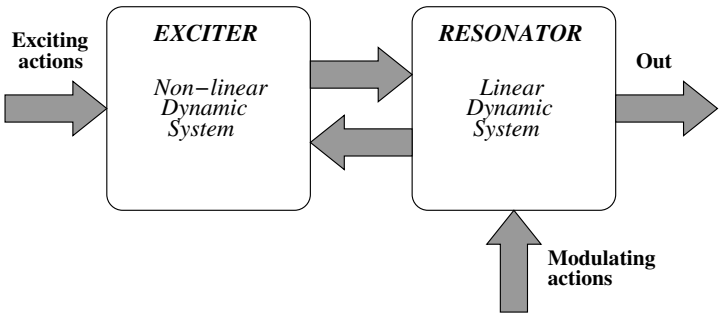


Figure 3.23: *Exciter-resonator interaction scheme for a musical instrument.*

where the oscillations actually take place (an acoustic bore, a string, a bar, etc) and is therefore related to such sound attributes as pitch, spectral envelope, and so on. The *exciter* controls the way energy is injected into the system, thus initiating and possibly sustaining the oscillations, and relates in particular to properties of attack transients. A simple yet striking demonstration of the effectiveness of the exciter/resonator schematization is provided by mounting a clarinet mouthpiece on a flute.[8] The bore boundary conditions are changed from open-open to closed-open, so that it plays one octave lower, and the resulting instrument is perceived as a bad sounding clarinet. In other words, the excitation mechanism defines sound identity ("it's a clarinet"), while the resonator is mostly associated to sound quality ("it's a *bad* clarinet").

The interaction between the two blocks is a two-way interaction, where the state of each block influences the other. As an example, the impact force between a hammer and a string depends on the displacements and velocities of both hammer and string, and affects both. Clearly there are also examples where non-linearities in the excitation are negligible: plucked string instruments can be conveniently treated as linear systems (strings and instrument body), where the "pluck" is described as a non-equilibrium initial condition (i.e., the pluck gives a string a non-zero displacement distribution and a null velocity distribution).

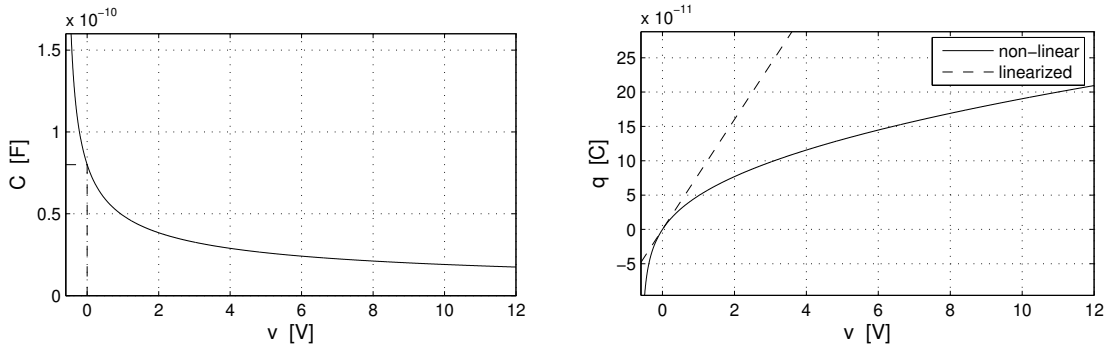
Finally, note that non-linearities are not necessarily related to excitation mechanisms only: even resonators, that are assumed to be linear in a first approximation, can exhibit non-linear behaviors. As an example, when a string vibrates outside the limit of small oscillations its length cannot be anymore assumed to be constant, but varies (together with string tension) during an oscillation cycle: this length- and tension-modulation mechanism can produce perceivable pitch glides in the sound. Similar considerations apply to other systems (e.g. non-linear circuit elements).

3.6.1 Non-linear circuits

3.6.1.1 Non-linear capacities

Consider the well known *Chua-Felderhoff* electrical circuit: this is a *RLC* circuit, made of a series connection of a resistor R, an inductor L and a capacitor C. The elements R and L are constant, while this is not the case for C. More precisely, the characteristic of the capacitance is a function of the voltage v,

[8] The author has enjoyed a live demonstration with such a "flarinet", performed by Joe Wolfe while giving a seminar in Venice, 2000.



(a) (b)

Figure 3.24: *Non-linear behavior of (a) capacitance $C(v)$ and (b) charge $q(v)$ in the Chua-Felderhoff circuit.*

so that the system is described as follows:

$$v(q) = \frac{1}{2v_0C_0^2}\left(q^2 + q\sqrt{q^2 + 4v_0^2C_0^2}\right), \quad \Leftrightarrow \quad C(v) = \frac{C_0}{\sqrt{1 + \dfrac{v}{v_0}}}, \tag{3.104}$$
$$v(q) + R\dot{q}(t) + L\ddot{q}(t) = v_e(t), \qquad (v > v_0).$$

The variable $q(t)$ stands for the charge on the capacitor, and $v_e(t)$ is an applied voltage. Note that $C(v) \sim C_0$ when $v \to 0$, i.e. the system is a linear *RLC* circuit in the limit of small oscillations. However, for larger voltage v this approximation does not hold, and $C(v)$, $q(v)$ behave as depicted in Fig. 3.24(a) and (b), respectively. There is no easy way to translate the non-linear relation (3.104) into the Laplace domain, because the definition of impedance given in Sec. 3.2.1 assumes linearity of the circuit elements.

The Chua-Felderhoff circuit has been extensively studied and is one of the classical systems used for exemplifying transition to chaotic behavior: when the peak of the voltage generator is increased, the behavior of the charge $q(t)$ on the capacitor undergoes successive bifurcations.

3.6.1.2 Vacuum tubes

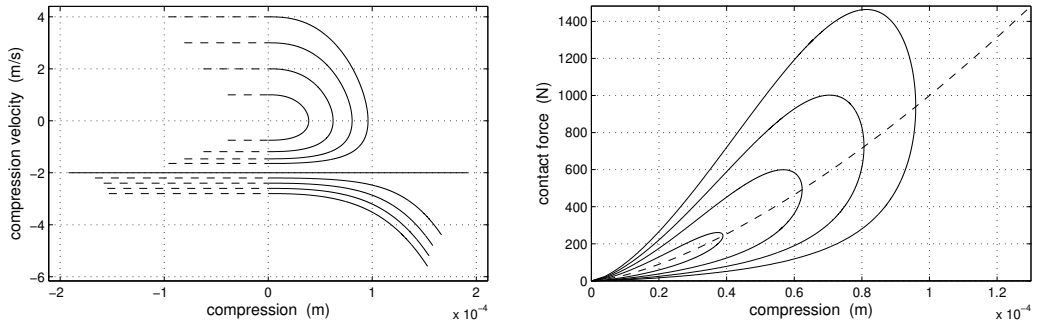
.....

3.6.2 Mechanical interactions

3.6.2.1 Impacts

Several musical and non musical classes of sounds are produced by a single impact of two objects, one of which (at least) resonates as a consequence of the collision. Moreover, impact is at the basis of other more complex mechanical contacts: as an example, scraping and rolling can be seen as temporal sequences of micro-impacts between non-smooth surfaces.

The ideal impact is a force signal shaped like a Dirac delta in time. It imparts to the resonator an ideal force impulse in an infinitesimal time. If the resonator is initially at rest, such force impulse imparts to the resonator initial conditions given by null initial displacement and a non-zero initial velocity whose magnitude depends on the magnitude of the delta.



(a) (b)

Figure 3.25: *The non-linear impact model (3.106): (a) phase portrait of a point mass hitting a hard surface; (b) the corresponding non-linear force during impact.*

In a less ideal impact model one would assume that the impact force is non-null over a finite duration or time (the *contact time* between the colliding objects) and takes finite values. The force magnitude is related to the impact energy (e.g. the impact velocity of the hammer hitting the resonator), while the contact time is related to the hardness of the impact. A simple signal model of the impact force is the following:

$$f(t) = \begin{cases} \frac{f_{\max}}{2}\left[1 - \cos(\frac{2\pi t}{\tau})\right], & 0 \leq t \leq \tau, \\ 0, & \text{otherwise}, \end{cases} \tag{3.105}$$

where τ is the contact time and $f_{\max}$ is the maximum force value.

More complex models must take into account other effects. There is dissipation of energy during contact. The contact force itself is a function of the relative compression $x(t)$ between the two contacting objects (which may be thought as the difference between the displacements of the two objects during the contact), and also of the compression velocity $v(t) = \dot{x}(t)$. Accordingly, a more physically-based model of the impact force is the following:

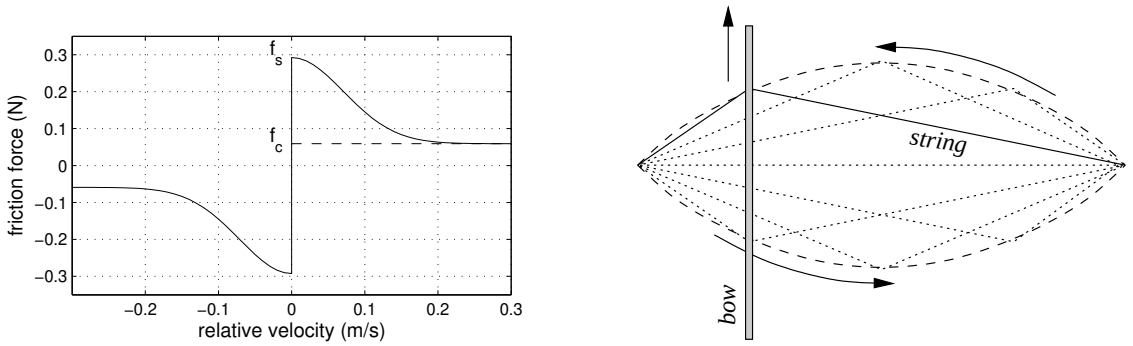
$$f(x(t), v(t)) = \begin{cases} kx(t)^{\alpha} + \lambda x(t)^{\alpha} v(t), & x > 0, \\ 0, & \text{otherwise}, \end{cases} \tag{3.106}$$

where k is the force *stiffness*, λ is the force damping weight, and the exponent α depends on the local geometry around the contact area. As an example, according to Herz theory of contact an ideal impact between two spherical objects obeys this equation with $\alpha = 3/2$ and $\lambda = 0$.

Figure 3.25(a) depicts the simulation of a point mass hitting a rigid surface with the impact model (3.106: the phase portrait shows that due to dissipation the mass velocity after the impact is always lower in magnitude than the initial impact velocity, and converges to a limit value. Figure 3.25(a) shows the corresponding impact force: it has a non-linear characteristics that depends on the exponent α, and it exhibits a hysteresis effect that is associated to to the dissipative component $\lambda x^{\alpha} v$. This plot is qualitatively resemblant of what one would observe by measuring the contact force during a real impact of a small mass againts a rigid surface.

M-3.17

Simulate a modal oscillator excited by the non-linear impact force (3.106).



(a) (b)

Figure 3.26: *Stick-slip friction: (a) example of parametrization of a kinetic (static) friction curve; (b) Helmoltz motion resulting from stick-slip ideal string-bow interaction.*

3.6.2.2 Stick-slip friction

Stick-slip friction is a second relevant mechanical interaction in sound production. A typical musical example is the interaction between bow and string in a violin. A non musical example is the sound produced by a finger rubbing on a moisty window or on a glass.

We know from physics that static friction is higher than dynamic friction: the simplest model assumes that the friction force is proportional to the normal force f_N between two contacting objects, but the coefficient of proportionality is higher if there is no relative motion and is lower if there is relative motion. More refined models define the coefficient of proportionality as a function of the relative velocity. These are called *kinetic* models (as the friction force is assumed to be a function of velocity only), or *static* models (since the force-velocity dependence is derived under stationary conditions). One possible parametrization of a kinetic friction force model is:

$$f(v(t)) = \begin{cases} \text{sgn}(v) \left[f_c + (f_s - f_c) e^{-(v/v_s)^2} \right], & f_N > 0, \\ 0, & \text{otherwise}, \end{cases} \tag{3.107}$$

where f_c, f_s are the Coulomb force and the stiction (short for static friction) force respectively, while v_s is named Stribeck velocity. The Coulomb force and the stiction force are related to the normal force through the equations $f_s = \mu_s f_N$ and $f_c = \mu_d f_N$, where μ_s and μ_d are the static and dynamic friction coefficients. If $f_N \leq 0$ this means that there is no contact. The dependence of the friction force on velocity, as given in Eq. (3.107), is shown in Fig. 3.26(a).

When two objects in relative motion interact through a friction force of this kind, a *stick-slip* phenomenon is generated in which the two objects remain in static contact for a certain amount of time (the "stick" phase) and suddenly detach (the "slip" phase). Sound generation occurs when this alternation of stick and slip phases occurs in an almost periodic fashion and with an audio rate, typically locked to some of the proper resonance frequencies of the interacting objects.

An example of stick-slip interaction is the *Helmoltz motion* occurring in an ideal, rigidly terminated, bowed string (see Fig. 3.26(b)). Assuming the bow to be perfectly rigid and to be in contact with the string in a single point, the string motion at the contact point is a sawtooth signal in which the string remains stuck to the bow hair for a considerable fraction of each vibratory cycle, and slips back abruptly when its displacement becomes large enough, to begin the next cycle. In normal playing condition the resulting frequency of oscillation is almost coincident with the first-mode frequency of the string. Further

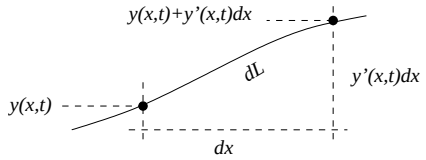


Figure 3.27: *Length dL of a string at point x over the segment dx.*

analysis of this Helmoltz motion would reveal that at every instant the shape of the string consists of two line segments joined by a corner, and the corner travels on an envelope composed of two parabolas.

M-3.18

Simulate a modal oscillator excited by the non-linear friction force (3.107).

More refined, *dynamic* friction models include some "memory". The dependence of friction on the relative sliding velocity is modeled using a differential equation. These models are able to take into account presliding behavior, where the friction force increases gradually for small displacement values. Static and dynamic friction models have the same behavior at high or stationary relative velocities, but dynamic models provide more accurate simulation of transients, which is particularly relevant for realistic sound synthesis.

3.6.2.3 Tension modulations

The phenomenon of tension modulation is qualitatively different from the previous examples. This non-linear effects is not generated from an external excitation force. It is a non-linear correction to the D'Alembert equation when the limit of small oscillations of the elastic medium is not valid.

The simplest example of tension modulation is encountered in a vibrating string with fixed ends. When the string is significantly displaced from equilibrium, its length and therefore also its tension are increased. When it returns closer to its equilibrium state, its length and tension are decreased. Clearly the rate of this tension modulation is twice the rate of the transversal vibration, since minimum tension occurs at equilibrium, and maximum tension occurs at both extreme diplacements.

$$\mu\frac{\partial^2 p}{\partial t^2}(x,t) - T[y(x,t)]\frac{\partial^2 p}{\partial x^2}(x,t) + EI\frac{\partial^4 p}{\partial^4 x}(x,t) + d_1\frac{\partial p}{\partial t}(x,t) - d_2\frac{\partial^3 p}{\partial t \partial^2 x}(x,t) = 0, \qquad (3.108)$$

where $T[y(x,t)]$ is the string tension and is now a function of the string displacement. More precisely, it is proportional to the string length, which in turns depends on $y(x,t)$. From the theorem of Pithagoras, the length dL at point x over the segment dx is (see Fig. 3.27) $dL[y(x,t)] = \sqrt{dx^2 + (y'(x,t)dx)^2}$. Then the total string length deviation ΔL from the length L_0 at equilibrium is

$$\Delta L[y(x,t)] = L[y(x,t)] - L_0 = \int_0^{L_0} dL[y(x,t)] - L_0 = \int_0^{L_0} \sqrt{1 + y'(x,t)^2}dx - L_0. \qquad (3.109)$$

Then the tension is

$$T[y(x,t)] = T_0 + \frac{EA\Delta L[y(x,t)]}{L_0}, \qquad (3.110)$$

where A is the string section.

Tension modulation in a waveguide model can be simulated by using all-pass filters with time-varying coefficients that account for length modulation. Tension modulation in modal synthesis can be simulated

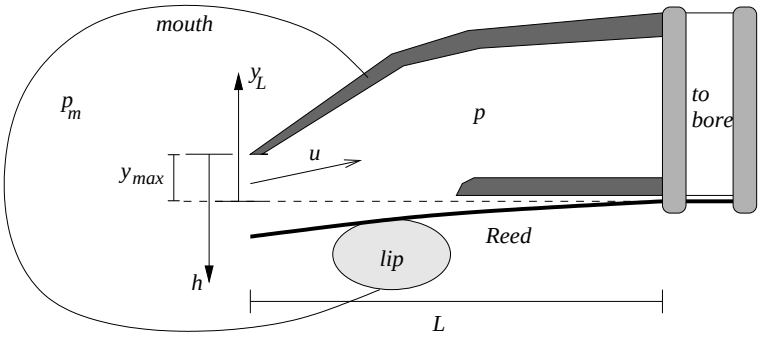


Figure 3.28: *Schematic representation of the reed-mouthpiece system.*

by finding the modes of Eq. (3.108). As an example, in the simple case where $d_2 = E = 0$ in Eq. (3.108), one can find the modes

$$\ddot{q}_i(t) + d_1\dot{q}(t) + i^2\left[\omega_0^2 + \omega_1^2\sum_{l=1}^{+\infty} l^2 q_l^2(t)\right] q_i(t) = 0, \qquad i = 1, \ldots, +\infty, \tag{3.111}$$

that can be interpreted as mechanical oscillators in which the frequency of oscillation depends on the modal displacement. In the general case of Eq. (3.108) including dispersion and frequency-dependent dissipation, a similar modal description can still be found.

3.6.3 Acoustic interactions

3.6.3.1 Jets

.....

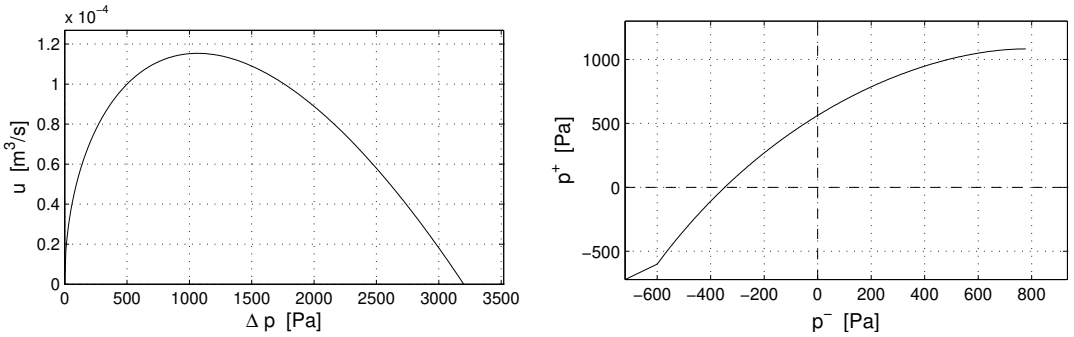
3.6.3.2 Quasi-static reeds

Reeds are acoustic valves that oscillate due to pressure differences at the two sides. The simplest example is the *single reed*, schematically represented in Fig. 3.28. The reed dimensions are small with respect to typical wavelengths in the resonator, thus pressure can be thought of as constant along the reed surfaces; under normal playing conditions, the first mode of oscillation of the reed is well above the main frequency components of the pressure signal in the resonator. Oscillations occur mainly in the vertical direction, and a single degree of freedom can be reasonably assumed, i.e. the vertical displacement y_L of the reed tip from the equilibrium. These considerations justify the choice of a *lumped* modeling approach for the reed.

The simplest posssible lumped model regards the reed as a system with stiffness only, neglecting inertia and damping properties. In this approximation the reed moves in phase with the pressure difference $\Delta p(t) = p_m(t) - p(t)$ across the reed:

$$k y_L(t) = S_d \Delta p(t) \qquad \Rightarrow \quad k_a y_L(t) = \Delta p(t), \tag{3.112}$$

where p_m is the pressure inside the performer's mouth, p is the (oscillating) acoustic pressure inside the instrument bore, k is the effective reed stiffness, S_d is an effective driving surface on which the pressure Δp acts, and $k_a = k/S_d$ is the stiffness per unit area. Equation (3.112) is called a *quasi-static approximation* since it can be determined experimentally in static conditions where a constant pressure



(a) (b)

Figure 3.29: *Quasi-static approximation of a single reed; (a) u versus Δp and (b) rotated mapping p^+ versus p^-.*

difference δp is injected into the system and the corresponding constant displacement y_L is measured after an initial transient.

As far as aerodynamics is concerned, the relation between the reed opening $h(t)$, the airflow $u(t)$ through the slit, and the pressure drop $\Delta p(t)$ can be approximated through the equation

$$\Delta p(t) = f(u(t), h(t)) = \mathrm{sgn}[u(t)] \frac{\rho_{\text{air}}}{2} \frac{|\, u(t)\,|^2}{w h(t)}, \tag{3.113}$$

where w is the reed width. This equation is derived from the Bernoulli law.[9] Using Eq. (3.112), the reed opening h is computed as $h = y_{max} - y_L = y_{max} - \Delta p / k_a$, and by substituting this relation into Eq. (3.113) one finds

$$u(t) = \begin{cases} w\,\mathrm{sgn}[\Delta p(t)] \left(y_{max} - \frac{\Delta p(t)}{k_a}\right) \sqrt{\frac{2|\,\Delta p(t)\,|}{\rho_{\text{air}}}}, & \Delta p < k_a y_{max}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.114}$$

Figure 3.29(a) shows the plot of the resulting relation between u and Δp. For low Δp values, u increases until a maximum is reached at $\Delta p = k_a y_{max}/3$. For higher Δp values, the flow starts to drop due to reed closure, and reaches the value $u = 0$ at $\Delta p = k_a y_{max}$. Beyond this value the reed is completely closed.

This non-linear map can be used to construct a quasi-static reed model. If wave variables $p^{\pm}$ are introduced in the cylindrical bore, i.e. $p = p^+ + p^-$ and $u = p^+ - p^-$, then these relations can be substituted into Eq. (3.114). As a consequence this non-linearity can be turned in a new one in which p^+ depends on p^- through a non-linear reflection function R_{nl}, i.e. $p^+ = R_{nl}(p^-)$. This is depicted in Fig. 3.29(b).

Despite its simplicity, the quasi-static model is able to capture the basic non-linear mechanisms of self-sustained oscillations in a single reed instrument. Due to its compactness and low number of parameters, this model has been also used for sound synthesis purposes.

M-3.19

[9] The Bernoulli law holds for incompressible non-viscous fluids in stationary conditions, and states the relation $u = A \cdot x \cdot \Delta p^{1/2} \mathrm{sgn}(\Delta p)$ between the flow u and the pressure difference Δp through an aperture of width x. Some authors adopt for the single reed the generalized equation $u = [A \cdot x \Delta p^{1/2}\,\mathrm{sgn}(\Delta p)]^{1/\alpha}$, with an experimentally determined value $\alpha = 3/2$.

Write a function that computes the pressure wave $p^+[n]$ reflected into the bore from the wave $p^-[n]$ arriving from the bore, according to the quasi-static model (3.114). Implement a quasi-static clarinet model in which the quasi-static reed is coupled to a waveguide cylindrical bore and driven by a mouth pressure signal p_m. The bell can be modeled as a low-pass filter, that radiates frequencies above its cut-off (typically around 1500 Hz) and reflects low frequencies back inside the bore.

M-3.19 Solution

Further refinements to this model should include propagation losses, fractional-delay filters in order to allow for fine tuning of the bore length, and acoustic holes modeled as scattering filters connected through 3-port junctions to the main waveguide structure.

3.6.3.3 Dynamic reeds

More refined reed models need to take into account the dynamics of the reed. A reasonably accurate description is obtained through a second-order mechanical oscillator, driven by the pressure drop Δp between mouth and mouthpiece:

$$\begin{cases} m\ddot{y}_L(t) + r\dot{y}_L(t) + k(y_L(t) - y_0) = S_d \Delta p(t), & y_L < y_{max}, \\ y_L(t) = y_m \quad \text{and} \quad \dot{y}_L(t) = 0, & y_L \geq y_{max}, \end{cases} \tag{3.115}$$

where m and r represent the reed mass and damping, while other parameters and variables are defined as before. The constant y_0 represents the reed displacement at rest.

This modeling approach is reasonable for the same reasons mentioned before: small reed dimensions compared to typical wavelengths in the resonator, reed oscillation mainly in the vertical direction, and high frequencies of the transversal reed modes (only the first mode is relevant). Note that in Eq. (3.115) the phenomenon of reed beating (i.e. complete closure of the reed) is here incorporated in the lumped model in a non-physical way, by imposing an ideal "stop" when the reed tip reaches its maximum allowed displacement y_m. Note also that the quasi static approximation examined in the previous section corresponds to approximating the transfer function of this system with its value at 0 frequency.

Another refinement amounts to taking into account an additional component affecting the total flow inside the instrument: the reed motion generates the flow $S_d\dot{y}_L(t)$, proportional to the reed tip velocity. If we now call u the flow inside the instrument and u_f the flow entering from the slit, these are related through the following equation:

$$u(t) = u_f(t) + u_r(t), \qquad \text{with} \quad u_r(t) = S_r\dot{y}_L(t). \tag{3.116}$$

Incorporating this dynamics into the model results in more convincing sound synthesis, especially as far as transients are concerned. Realistic effects can be obtained, such as transitions to high regimes of oscillation. Both the resonance and the damping of the reed oscillator (3.115) and g play a role in helping transition to the second register of a single reed instrument. As an example, the clarion register in the clarinet plays a twelfth above the fundamental register and is usually obtained with the aid of a register hole. Howver the clarion register can be produced also without opening the register hole if the reed resonance matches a low harmonic of the playing frequency and the damping is small enough. Another playing regime in single reed instruments is the so-called reed regime ("squeaks"): this can be obtained by imparting an extremely low damping to the reed oscillator, so that the oscillation is governed by the reed resonance.

M-3.20

Implement a dynamic clarinet model in which the dynamic reed is coupled to a waveguide cylindrical bore and driven by a mouth pressure signal p_m.

Finally, additional degrees of freedom must be taken into account when simulating other types of reeds. *Double reeds* (such as those found in oboe and bassoon) are composed of two reeds that oscillate independently, and even if one assumes perfect symmetry of oscillation the flow model differs from the one examined previously, due to the smallness of the aperture. In so-called *lip reeds* the role of the reed is taken by the performer's lips, that are constrained into the mouthpiece and vibrate at the fundamental frequency: at least two degrees of freedom are needed to simulate lip vibration.

3.6.4 Computability issues

We have examined in Sec. 3.5.1 the concept of delay-free loop in the case of linear systems, and have mentioned some strategies for dealing with it. However, more severe computability problems can arise when simulating non-linear elements.

3.6.4.1 Non-linear systems and delay-free loops

It should be clear that in the non-linear case one cannot perform a rearrangement such as in (3.79), because a non-linear equation is not always analytically invertible. The question is then how to deal with the delay-free loop problem in the non-linear case.

One can use an *explicit* numerical method, that produces a system of difference equations in which there are no delay-free loops. This choice solves the computational problem but can introduce more severe artifacts in the numerical system: explicit methods have lower orders of accuracy with respect to implicit methods, and more importantly are not unconditionally stable, i.e. are not stable for any sampling frequency F_s and for any values of the system parameters.

A rudimentary solution, that is nonetheless often encountered in the literature of physical modeling, amounts to inserting fictitious delay elements z^{-1} in the computational scheme. In practice this is a variant of the previous approach: instead of using an explicit method from the beginning, one makes the computation explicit *a posteriori*, through the insertion of delay elements. While this "trick" can be acceptable at significantly high sampling rates, the insertion of delay elements can again deteriorate the accuracy and stability properties of the numerical system. Even worse, in this case one cannot determine analytically the stability range of the system.

3.6.4.2 Iterative methods

Numerical analysis provides iterative methods to find solutions of non-linear systems of algebraic equations: examples of such methods include fixed-point iteration and Newton iteration, and each of them requires specific hypothesis on the non-linear system to hold.

Using an iterative solver is advantageous over the previous approaches in that one can exploit the accuracy and stability properties of an implicit method without introducing additional numerical errors in the system. One major drawback, however, is that one does not know in advance the number of iterations that are needed for the solver to converge to the solution $\boldsymbol{y}[n]$: this can be a problem for real-time applications, where one wants to know the time needed to compute one sound sample.

See [Fontana and Avanzini, 2008] for details about Newton-Raphson and fixed-point iteration for the simulation of non-linear systems.

3.6.4.3 Sheared non-linearities

In many practical cases the delay-free loop problem takes the form of the implicit dependence

$$\boldsymbol{y}[n] = \boldsymbol{f}(\tilde{\boldsymbol{x}}[n] + \boldsymbol{K}\boldsymbol{y}[n]), \tag{3.117}$$

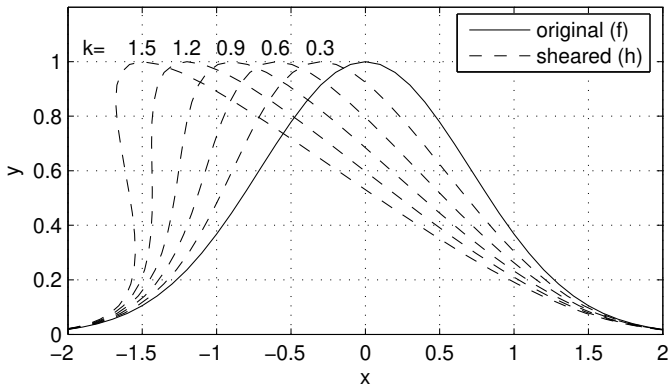


Figure 3.30: *Shear transformation of $f(x) = e^{-x^2}$ for various k values.*

where $\boldsymbol{f}$ is a non-linear function, and $\tilde{\boldsymbol{x}}[n]$ is a vector of variables that are known at time n. The variables $\boldsymbol{y}[n]$ depend instantaneously onto themselves in the above equation. If one could turn this implicit dependence into a new explicit dependence $\boldsymbol{y}[n] = \boldsymbol{h}(\tilde{\boldsymbol{x}}[n])$, this would solve the delay-free loop problem.

This is achieved using the *implicit mapping theorem*. Define the function $\boldsymbol{g}$ as

$$\boldsymbol{g}(\tilde{\boldsymbol{x}}, \boldsymbol{y}) = \boldsymbol{f}(\tilde{\boldsymbol{x}} + \boldsymbol{K}\boldsymbol{y}) - \boldsymbol{y}, \tag{3.118}$$

and assume that there is a point $(\tilde{\boldsymbol{x}}_0, \boldsymbol{y}_0)$ such that $\boldsymbol{g}(\tilde{\boldsymbol{x}}_0, \boldsymbol{y}_0) = 0$. Moreover, assume that the following condition holds

$$\det[\boldsymbol{J}_{\boldsymbol{y}}(\boldsymbol{g})(\tilde{\boldsymbol{x}}_0, \boldsymbol{y}_0)] = \det\left[\frac{g_i}{y_j}(\tilde{\boldsymbol{x}}_0, \boldsymbol{y}_0)\right]_{i,j} \neq 0, \tag{3.119}$$

where $\boldsymbol{J}_{\boldsymbol{y}}(\cdot)$ denotes the Jacobian matrix with respect to the $\boldsymbol{y}$ variables. From the definition of $\boldsymbol{g}$, it is seen that $\boldsymbol{J}_{\boldsymbol{y}}(\boldsymbol{g}) = \boldsymbol{J}_{\boldsymbol{x}}(\boldsymbol{f})\boldsymbol{K} - \boldsymbol{I}$. Therefore, condition (3.119) implies that the matrix $[\boldsymbol{J}_{\boldsymbol{x}}(\boldsymbol{f})\boldsymbol{K} - \boldsymbol{I}]$ must be non-singular at the point $(\tilde{\boldsymbol{x}}_0, \boldsymbol{y}_0)$. If these conditions are fulfilled, then the implicit mapping theorem states that a function $\boldsymbol{h}(\tilde{\boldsymbol{x}})$ exists locally (i.e. for points $\tilde{\boldsymbol{x}}$ in a neighborhood of $\tilde{\boldsymbol{x}}_0$), with the properties

$$\boldsymbol{h}(\tilde{\boldsymbol{x}}_0) = \boldsymbol{y}_0 \qquad \text{and} \quad \boldsymbol{g}(\tilde{\boldsymbol{x}}, \boldsymbol{h}(\tilde{\boldsymbol{x}})) = 0. \tag{3.120}$$

If the above conditions are fulfilled globally rather than in a neighborhood of $(\tilde{\boldsymbol{x}}_0, \boldsymbol{y}_0)$, then $\boldsymbol{h}$ is defined globally.

A few geometrical considerations can help understanding the shape of the new function $\boldsymbol{h}$. Consider the coordinate transformation

$$\begin{bmatrix} \tilde{\boldsymbol{x}} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \mathbb{I} & -\boldsymbol{K} \\ \boldsymbol{0} & \mathbb{I} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}. \tag{3.121}$$

This defines a *shear* that leaves the $\boldsymbol{y}$ axes unchanged and distorts the $\boldsymbol{x}$ axis into the $\tilde{\boldsymbol{x}}$ axis. The plot of the function $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$ "lives" in the $(\boldsymbol{x}, \boldsymbol{y})$ space. Then the plot of $\boldsymbol{y} = \boldsymbol{h}(\tilde{\boldsymbol{x}})$ is obtained by applying the coordinate transformation (3.121), and is therefore a sheared version of the former.

In order to understand this shear effect, consider the following example with a scalar function $f : \mathbb{R} \to \mathbb{R}$:

$$y[n] = f\left(x[n]\right) = e^{-(x[n]^2)}, \qquad \text{with} \quad x[n] = \tilde{x}[n] + ky[n]. \tag{3.122}$$

Condition (3.119) translates in this case in the condition $f'(x) \neq 1/k$, which has a straightforward geometrical interpretation: the shear transformation defined in Eq. (3.121) is such that the vector $[x, y]^T =$

$[k, 1]^T$ (i.e. a point with tangent $1/k$) is transformed into the vector $[\tilde{x}, y]^T = [0, 1]^T$ (i.e. a point with vertical tangent). This explains why the derivative of f cannot equal $1/k$.

Figure 3.30 shows the original function $f(x)$, together with the sheared one $h(\tilde{x})$, for various k values. It can be seen that the horizontal coordinate is distorted when applying the shearing transformation. Moreover, note that for $k = 1.5$ the new function $h(\tilde{x})$ cannot be defined globally, because the condition $f'(x) \neq 1/k$ is not fulfilled globally in this case.

M-3.21

Simulate a modal oscillator excited by the non-linear impact force $f(x(t) = kx(t)^\alpha$ (i.e. the impact model (3.106) with $\lambda = 0$) as follows: use an implicit numerical scheme (e.g. the bilinear transform), find the implicit dependence in the form (3.117), and construct the corresponding sheared non-linear function.

3.7 Commented bibliography

Sound modeling techniques can be classified according to many criteria. Two general references that address these issues are [De Poli, 1991, Smith III, 1991]. Specifically, the taxonomy based on *signal models* and *source models*, and their subclasses, proposed at the beginning of this chapter is based on [De Poli, 1991].

Seminal ideas that eventually lead to the definition of physically-based sound modeling techniques are to be found in research on musical instrument acoustics. Some classic papers in this area are [Hiller and Ruiz, 1971a,b, Schumacher, 1981, McIntyre et al., 1983]. In particular, the two citations at the beginning of the Introduction are taken from Hiller and Ruiz [1971a], McIntyre et al. [1983], respectively. A book that covers the topic of musical acoustics exhaustively is [Fletcher and Rossing, 1991]. In particular our discussion of the analogies between electrical, mechanical, and acoustic systems, given in Sec. 3.2.1 is based on an analogous discussion in [Fletcher and Rossing, 1991, Ch.1].

A general overview on approaches and techniques used in physical modeling, with an emphasis on structural and computational aspects, is provided by De Poli and Rocchesso [1998]. Figure 3.23 in this chapter (typical block scheme of a musical instrument model) is based on an analogous scheme in [De Poli and Rocchesso, 1998]. Two more recent and very complete tutorial papers on the topic of physical modeling are [Smith III, 2004] and [Välimäki et al., 2006].

About waveguide modeling approaches. The theory of 1-D waveguide models is now well established. An exhaustive introduction to the topic is given by Smith III [1998], who provides full derivations of waveguide structures and examples of musical instrument modeling, together with a vast bibliography. A more recent and even more exhaustive overview is given by the same author in [Smith III, 2008].

The basic principles of waveguide models were already present in the work of Kelly and Lochbaum [1962] on speech synthesis, where a so-called "transmission-line modeling" approach was used to simulate the human vocal tract through delay lines and scattering junctions. The definition of "digital waveguide modeling" was introduced later by Smith III [1985] in the context of musical applications, because of an analogy to the concept of waveguide that has been used, for example, in microwave technology. The Karplus-Strong algorithm, which we have regarded as the first step toward the development of digital waveguide structures, was originally proposed by Karplus and Strong [1983]. Fractional-delay filters: detailed discussion is provided by Laakso et al. [1996]. Modeling of dissipation and dispersion: Bank [2006] discusses the topic at length, with application to physically-based synthesis of the piano. In particular, the frequency-dependent dissipation model reported in Eq. (3.48) was first proposed by Bensa et al. [2003], although in the context of finite-difference simulations, as an improvement of the dissipation model by Hiller and Ruiz [1971a]. About waveguide junctions. Many textbooks on digital speech processing discuss multitube lossless models of the vocal tract, which are basically cylindrical waveguide sections connected by Kelly-Lochbaum junctions: see e.g. [Deller et al., 1993]. We have not addressed

the topic of higher dimensional (2- and 3-D) waveguide structures: seminal ideas were first presented by van Duyne and Smith III [1993].

About lumped modeling approaches. Numerical and computational aspects: most of the techniques described in Sec. 3.5.1 are found in DSP textbooks: see e.g. [Mitra, 2005]. In the field of numerical analysis, a comprehensive discussion on numerical methods for ordinary differential equations is given by Lambert [1993]. The example illustrated in Fig. 3.17 about delay-free computational paths in linear systems is adapted from [Mitra, 2005, Sec. 6.1.3, Fig. 6.5]. A classic reference to the theory of Wave Digital Filters (*WDF*) theory is [Fettweis, 1986].

Finite difference schemes have been applied to also to the explicit numerical simulation of partial differential equations, e.g. for modeling idiophones [Chaigne and Doutaut, 1997] and single reed systems [Stewart and Strong, 1980]. A recent book about the applications of finite difference methods to numerical sound synthesis is [Bilbao, 2009], which discusses the fundamentals of finite differences and shows how they can be employed to simulate strings, bars, plates, membranes, acoustic tubes. Among other lumped modeling approaches, in the early nineties Cadoz and coworkers have introduced the CORDIS-ANIMA model [Florens and Cadoz, 1991], which describes vibrating bodies as a set of interconnected mass-spring-damper cells.

Modal synthesis. A classic presentation of modal synthesis techniques is [Adrien, 1991]. Cook [1997] developed a series of "physically-informed" approaches to the modeling of percussion sounds, which are based on a modal description. The use of modal sound synthesis to virtual reality applications is discussed in [van den Doel and Pai, 2004]. A corpus of relevant contributions in this field has been provided by Rabenstein and coworkers [Trautmann and Rabenstein, 2003], who have proposed the so-called functional transformation method (FTM): in essence, the method exploits the existence of an analytical form of the modal parameters for a set of relevant multidimensional differential systems, including strings and membranes with various boundary conditions. Our examples of modal analysis for simple 1-D and 2-D shapes is based on [Fletcher and Rossing, 1991, Ch.2-3]. The same book also shows experimental results of modal analysis on several musical instruments, including modal shapes and Chladni patterns. In addition to linear prediction techniques and partial tracking methods, already discussed in Chapter *Sound modeling: signal based approaches*, a method for high-resolution estimate of modal parameters from sound analysis has been proposed in [Esquef et al., 2003].

About non-linear physical models. The non-linear impact model of Eq. (3.106) was first proposed by Hunt and Crossley [1975]. . Concerning stick-slip friction models, an overview of traditional models in the context of sound synthesis applications (bowed strings) is provided by Serafin [2004]. More complex dynamic stick-slip models, typically used in the literature of automatic control, have been recently applied to sound synthesis by Avanzini et al. [2005]. We have seen that the reed mechanism is that of pressure-controlled valves: a classic paper on the topic is [Fletcher, 1993]. The quasi-static single reed examined in Sec. 3.6.3 was first studied by Schumacher [1981] and has been used extensively in the literature. Other types of reeds: for the double reed see [Guillemain, 2004], for the lip reed see [Adachi and aki Sato, 1996]. Lip reeds have some similarities with vocal fold functioning: a classic example of a vocal fold model applied to voice synthesis is [Ishizaka and Flanagan, 1972].

We have seen that new problems are encountered when non-linear elements are present in the delay-free computational path: Borin et al. [2000] provides a discussion of these issues, together with a proposed non-iterative solution (in brief, a set of hypotheses and techniques to pre-compute a "sheared" non-linear function that makes the numerical scheme computable), and applications to the simulation of acoustic systems.

References

Seiji Adachi and Masa aki Sato. Trumpet Sound Simulation Using a Two-dimensional Lip Vibration Model. *J. Acoust. Soc. Am.*, 99(2):1200–1209, Feb. 1996.

Jean-Marie Adrien. The missing link: Modal synthesis. In Giovanni De Poli, Aldo Piccialli, and Curtis Roads, editors, *Representations of Musical Signals*, pages 269–297. MIT Press, Cambridge, MA, 1991.

Federico Avanzini, Stefania Serafin, and Davide Rocchesso. Interactive simulation of rigid body interaction with friction-induced sound generation. *IEEE Trans. Speech Audio Process.*, 13(6):1073–1081, Nov. 2005.

Balasz Bank. *Physics-based Sound Synthesis of String Instruments Including Geometric Nonlinearities*. PhD thesis, Budapest University of Technology and Economics, Dep. of Measurement and Information Systems, Budapest, 2006.

Julien Bensa, Stefan Bilbao, Richard Kronland-Martinet, and Julius O. Smith III. The simulation of piano string vibration: From physical models to finite difference schemes and digital waveguides. *J. Acoust. Soc. Am.*, 114(2):1095–1107, Aug. 2003.

Stefan Bilbao. *Numerical sound synthesis - Finite difference schemes and simulation in musical acoustics*. John Wiley & Sons, Chichester, 2009.

Gianpaolo Borin, Giovanni De Poli, and Davide Rocchesso. Elimination of delay-free loops in discrete-time models of nonlinear acoustic systems. *IEEE Trans. Speech Audio Process.*, 8(5):597–606, Sep. 2000.

Antoine Chaigne and Vincent Doutaut. Numerical Simulations of Xylophones. I. Time-domain Modeling of the Vibrating Bar. *J. Acoust. Soc. Am.*, 101(1):539–557, Jan. 1997.

Perry R. Cook. Physically informed sonic modeling (PhISM): Synthesis of percussive sounds. *Computer Music J.*, 21(3): 38–49, 1997.

Giovanni De Poli. A Tutorial on Digital Sound Synthesis Techniques. In Curtis Roads, editor, *The Music Machine*, pages 429–447. MIT Press, 1991.

Giovanni De Poli and Davide Rocchesso. Physically Based Sound Modelling. *Organized Sound*, 3(1):61–76, Apr. 1998.

John R Deller, John G. Proakis, and John. H.L. Hansen. *Discrete-Time Processing of Speech Signals*. Macmillan, New York, 1993.

Paulo A. A. Esquef, Matti Karjalainen, and Vesa Välimäki. Frequency-zooming arma modeling for analysis of noisy string instrument tones. *EURASIP Journal on Applied Signal Processing*, 2003(10):953–967, 2003.

Alfred Fettweis. Wave Digital Filters: Theory and Practice. *Proceedings of the IEEE*, 74(2):270–327, Feb. 1986.

Neville H. Fletcher. Autonomous Vibration of Simple Pressure-Controlled Valves in Gas Flows. *J. Acoust. Soc. Am.*, 93(4): 2172–2180, Apr. 1993.

Neville H. Fletcher and Thomas D. Rossing. *The physics of musical instruments*. Springer-Verlag, New York, 1991.

Jean Luc Florens and Claude Cadoz. The physical model: modeling and simulating the instrumental universe. In Giovanni De Poli, Aldo Piccialli, and Curtis Roads, editors, *Representations of Musical Signals*, pages 227–268. MIT Press, Cambridge, MA, 1991.

Federico Fontana and Federico Avanzini. Computation of delay-free nonlinear digital filter networks. Application to chaotic circuits and intracellular signal transduction. *IEEE Trans. Sig. Process.*, 56(10):4703–4715, Oct. 2008.

Philippe Guillemain. A digital synthesis model of double-reed wind instruments. *EURASIP Journal on Applied Signal Processing*, 2004(1):990–1000, Jan. 2004.

Lejaren Hiller and Paul Ruiz. Synthesizing Musical Sounds by Solving the Wave Equation for Vibrating Objects: Part I. *J. Audio Eng. Soc.*, 19(6):462–470, June 1971a.

Lejaren Hiller and Paul Ruiz. Synthesizing Musical Sounds by Solving the Wave Equation for Vibrating Objects: Part II. *J. Audio Eng. Soc.*, 19(7):542–551, July 1971b.

Kenneth H. Hunt and F. R. Erskine Crossley. Coefficient of restitution interpreted as damping in vibroimpact. *ASME J. Applied Mech.*, 42:440–445, June 1975.

Kenzo Ishizaka and James L. Flanagan. Synthesis of voiced sounds from a two-mass model of the vocal cords. *Bell Syst. Tech. J.*, 51:1233–1268, 1972.

Kevin Karplus and Alexander Strong. Digital Synthesis of Plucked String and Drum Timbres. *Computer Music J.*, 7(2):43–55, 1983.

John L. Kelly and Carol C. Lochbaum. Speech synthesis. In *Proc. 4th Int. Congr. Acoustics*, pages 1–4, Copenhagen, Sep. 1962.

Timo I. Laakso, Vesa Välimäki, Matti Karjalainen, and Unto K. Laine. Splitting the Unit Delay Tools for Fractional Delay Filter Design. *IEEE Signal Processing Magazine*, 13(1):30–60, Jan. 1996.

John D. Lambert. *Numerical Methods for Ordinary Differential Systems*. John Wiley & Sons, 1993.

Michael E. McIntyre, Robert T. Schumacher, and James Woodhouse. On the Oscillations of Musical Instruments. *J. Acoust. Soc. Am.*, 74(5):1325–1345, Nov. 1983.

Sanjit K Mitra. *Digital Signal Processing*. McGraw-Hill, third edition, 2005.

Robert T. Schumacher. *Ab Initio* Calculations of the Oscillations of a Clarinet. *Acustica*, 48(2):71–85, 1981.

Stefania Serafin. *The sound of friction: real-time models, playability and musical applications*. PhD thesis, Stanford University, Center for Computer Research in Music and Acoustics, Stanford, 2004.

Julius O. Smith III. A new approach to digital reverberation using closed waveguide networks. In *Proc. Int. Computer Music Conf. (ICMC'85)*, pages 47–53, Vancouver, 1985.

Julius O. Smith III. Viewpoints on the History of Digital Synthesis. In *Proc. Int. Computer Music Conf. (ICMC'91)*, pages 1–10, Montreal, Oct. 1991.

Julius O. Smith III. Principles of digital waveguide models of musical instruments. In Mark Kahrs and Karl-Heinz Brandenburg, editors, *Applications of Digital Signal Processing to Audio and Acoustics*, pages 417–466. Kluwer Academic Publishers, New York, Mar. 1998.

Julius O. Smith III. Virtual acoustic musical instruments: Review and update. *Journal of New Music Research*, 33(3):283–304, Autumn 2004.

Julius O. Smith III. *Physical Audio Signal Processing: for Virtual Musical Instruments and Digital Audio Effects, December 2008 Edition*. `http://ccrma.stanford.edu/~jos/pasp/`, 2008. Accessed 15/12/2008.

Stephen E. Stewart and William J. Strong. Functional Model of a Simplified Clarinet. *J. Acoust. Soc. Am.*, 68(1):109–120, July 1980.

Lutz Trautmann and Rudolf Rabenstein. *Digital Sound Synthesis by Physical Modeling Using the Functional Transformation Method*. Kluwer Academic, New York, 2003.

Vesa Välimäki, Jyri Pakarinen, Cumhur Erkut, and Matti Karjalainen. Discrete-time modelling of musical instruments. *Rep. Prog. Phys.*, 69(1):1–78, 2006.

Kees van den Doel and Dinesh K. Pai. Modal Synthesis for Vibrating Objects. In Ken Greenebaum, editor, *Audio Anecdotes*. AK Peters, Natick, MA, 2004.

S. A. van Duyne and J. O. Smith III. The 2-D Digital Waveguide Mesh. In *Proc. IEEE Workshop on Applications of Sig. Process. to Audio and Acoustics (WASPAA'93)*, pages 177–180, New Paltz (NY), Oct. 1993.

Contents