# Musical instrument modeling: the case of the piano

F. Avanzini[a], B. Bank[a,b], G. Borin[a,c], G. De Poli[a], F. Fontana[c], and D. Rocchesso[c]

[a]Dip. di Elettronica ed Informatica - Università degli Studi di Padova, Italy

[b] Dept. of Measurement and Information Systems, Budapest Univ. of Technology and Economics, Hungary

[c]Dip. di Informatica - Università degli Studi di Verona, Italy

avanzini@dei.unipd.it, bank@mit.bme.hu, borin@prn.it,

depoli@dei.unipd.it, fontana@sci.univr.it, rocchesso@sci.univr.it

## Abstract

Real-time sound synthesis by physical modeling requires accurate design of each model block, together with special care on efficiency, computability, complexity issues. This paper review the case of the piano: implementation of the complete model is discussed, from the sound generation mechanism to radiation issues and coupling-pedal effects. Several design techniques are discussed and compared with focus on accuracy and efficiency issues.

## 1  Introduction

Sounds produced by acoustic musical instruments are ultimately caused by the physical vibrations of mechanical resonators or air volumes. These can be described at a *signal* level, where only the time-evolution of the acoustic pressure is considered and no assumptions on the generation mechanism are made; alternatively, *source* models can be developed, that are based on a physical descriptions of the sound production processes [1, 2]. Such an approach can be useful both for gaining a better insight in the functioning of the instruments and for designing sound synthesis algorithms; however, while complicated and accurate models have to be used for understanding physical phenomena, efficient sound synthesis calls for fast algorithms. Therefore a trade-off between accuracy and simplicity of the description has always to be found.

Physics-based synthesis algorithms provide *semantic* sound representations, since the control parameters have a straightforward physical interpretation in terms of masses, springs, dimensions and so on. Thus, modification of the parameters leads in general to meaningful results, and can help the user in interacting with the virtual instrument. Indeed, source models of sounding objects (not necessarily musical instruments) are nowadays gaining popularity in the multimedia community, due to their potential applications in human-computer interaction and the easiness in synchronizing audio and visual synthesis [3].

In this paper we review some of the strategies and algorithms of physical modeling, with special reference to piano simulation. This is a particularly interesting instrument, both for its prominence in western music and for the complexity of its functioning [4]. The models described here are all based on digital waveguides, since these have been found to be the most appropriate for real-time applications [5, 6]. As early as 1987, Garnett [7] presented a physical waveguide piano model. There a semi-physical lumped hammer model is connected to the waveguide string, which incorporates allpass filters for simulating dispersion and first order FIR filters for modeling losses. The bridge is treated as a common termination, where all the strings are connected. The soundboard is modeled by a set of waveguides, all connected to the same termination. The only imperfection here lies in the hammer model, since hammer velocity cannot be mapped to sound in a physical way. However, as we shall see in the next sections, the main ideas on string and soundboard modeling remained almost the same up to present days.

In 1995, Van Duyne and Smith presented a model based on commuted synthesis [8]; in this approach, a waveguide string model is fed using an excitation table; moreover, the radiation properties of the soundboard is *commuted*, i.e. its response is included in the excitation table. Although extremely efficient, commuted synthesis has some drawbacks, namely dynamic behavior in response to the player's action is difficult to tune: for instance, repeated strikes on a vibrating string are not easily modeled.

In order to account for more realistic behavior, a fully physical description of the excitation mechanism has to be developed. Early results in hammer modeling were obtained in 1990 by Borin *et al.* [9]. As part of a collaboration between the University of Padova and General-music, Borin *et al.* [10] presented a complete real-time model already in 1997; the hammer is treated as a lumped model, where a mass is connected in parallel with a non-linear spring that accounts for the felt compression characteristics; the string is simulated with a waveguide structure, and coupling between strings and with the soundboard are treated by connecting all the strings to a single lumped load. This research produced a number of byproducts, such as physically-based piano effects (pedal and damper), as well as electro-mechanical piano models implemented in commercial keyboards.

In 2000, Bank [11] introduced a similar physical model, with the same functional blocks but with a somewhat different implementation. An alternative approach was used for the solution of the hammer differential equation. independent string models were used without any coupling, and the influence of the soundboard on decay times was taken into account by high-order loss filters. Beating and

two-stage decay of the piano sound was modeled by using a resonator bank in parallel with the basic string model. The use of feedback delay networks was suggested for simulating the effects of soundboard radiation.

The remaining of this paper addresses the design of each element of a piano model (i.e. hammer, string and soundboard). Discussion is carried on with particular emphasis to real-time applications, where the time complexity of algorithms plays a key role. Perceptual issues are also addressed, since a precise knowledge of what is significant to the human ear can drive the degree of accuracy of the design. Section 2 deals with general aspects of piano acoustics, and points out the most relevant features of piano tones spectra. In Sec. 3 the hammer is discussed, and efficient numerical techniques are presented that allow to overcome non-computability problem in the non-linear discretized system. Section 4 is devoted to string modeling and simulation of losses, dispersion and fractional delays; dispersion is the most demanding problem in terms of computation, and is discussed from both the filter design and the perceptual viewpoints. Finally, Sec. 5 deals with the soundboard, which is to a large extent still an open problem; diverse techniques for its simulation are described and future directions in research are outlined.

## 2 General considerations

### 2.1 Spectral complexity

Piano sounds are the final product of a complex synthesis process, involving all the instrument body. As a result of this complexity, each piano note exhibits its unique sound features and nuances, especially in high quality instruments. Moreover, just varying the impact force on a single key allows the player to explore a rich dynamic space. For these reasons, many cheap solutions which were adopted in the earlier electronic pianos inevitably resulted in poor sound quality, due to low hardware capabilities (e.g. groups through resampling of one single prototype note or dynamics through changes in loudness).

This uniqueness is highlighted even looking at steady-state spectral analysis of different piano sounds [12]. As an example, Fig. 1 shows steady-state spectra of two notes, $C_2$ and $C_6$ respectively, recorded in an anechoic chamber from a Steinway C Grand piano. Each plot presents spectra generated from (above) a *fortissimo* (*ff*) note and (below) a *pianissimo* (*pp*) note. The spectra of the *pp* notes have been lowered of 30 dB to avoid superposition with the *ff* plots. In the case of the $C_2$ note it can be seen that, apart from the different loudness, the *ff* has an audible spectral content up to 6 kHz, whereas the components of the *pp* note become negligible over 3 kHz. In any case, substantial differences in the spectra start from 800 Hz, suggesting the presence of important non-linear effects happening when a piano note varies in its dynamic. The case of the $C_6$ note is even more interesting. The presence of the above-mentioned effects originates four new steady-state spectral components in the *ff*.
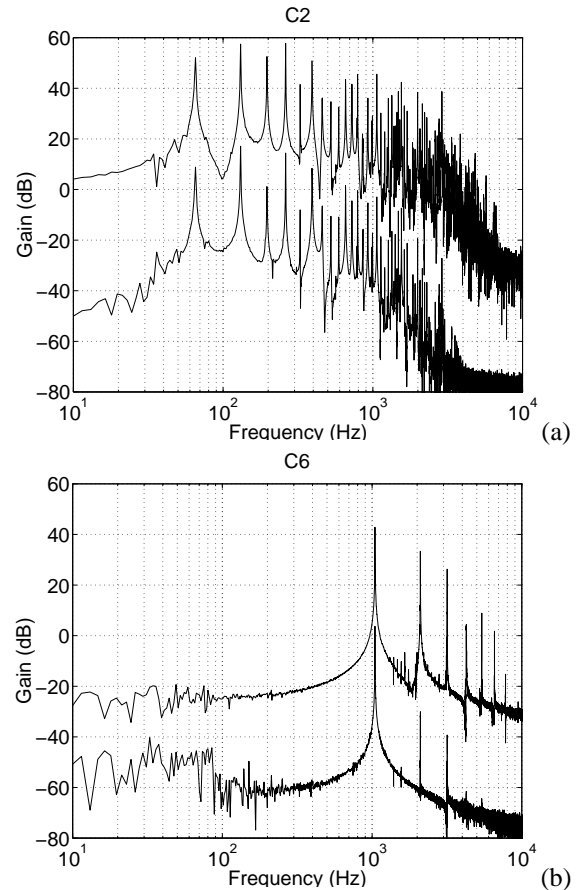


Figure 1: Spectra generated from (above) a *fortissimo* (*ff*) note and (below) a *pianissimo* (*pp*) note; (a) $C_2$ string and (b) $C_6$ string.

These examples are far from describing the whole complexity of variations occurring during a change in the note or in the dynamic. For instance, the initial transients are not accounted by a steady-state spectral analysis. Nevertheless, they give an idea of the complex results of the sound synthesis process in the piano, and how difficult is understanding which sound feature is caused by what. Accounting for such dynamic variations in a wavetable electronic piano is not trivial: dynamic post-processing filters can be designed, that shape the spectrum according to key velocity, but finding a satisfactory mapping from velocity to filter response in far from being an easy task. Alternatively, a physical model can be developed, that mimics as closely as possible the acoustics of the instrument; this is the topic of the next sections.

### 2.2 Acoustics and model structure

The general structure of the piano is the following: an iron frame is attached to the upper part of the wooden case and the strings are extended upon this in a direction nearly perpendicular to the keyboard. That end of the string which is closer to the keyboard is connected to the tuning pins on the bin block, and the other end, after crossing the bridge, is attached to the hitch-pin rail of the frame. The bridge is a thin wooden bar transmitting the
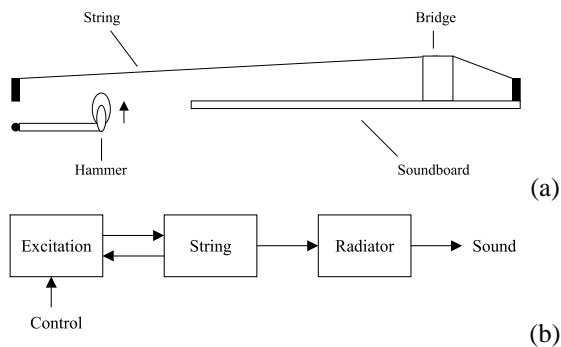
Figure 2: General structures; (a) schematic representation of the instrument; (b) model structure.

vibration of the string to the soundboard, which can be found under the frame. This is displayed in Fig. 2(a).

According to the above-mentioned parts, the sound-production mechanism of the piano can be divided into three steps. The first is the excitation, the hammer strike. The kinetic energy taken in by the player is transformed to kinetic energy of the hammer, which hits the string and transforms it to vibrational energy. This is stored by the string in its normal modes; some is dissipated due to internal losses, the remaining gets to the soundboard through the bridge. The soundboard converts the vibrational energy to acoustical energy, the audible sound. The complex spectra depicted in Fig. 1 are the result of this process.

Since the physical modeling approach tries to simulate the structure of the instrument and not the sound itself, the blocks in the piano model resemble the parts of a real piano. The structure is displayed in Fig. 2(b). The first model block is the excitation, the hammer strike. Its output propagates to the string, which determines the fundamental frequency of the tone. The quasi-periodic output signal is filtered through a post-processing block, covering the radiation effects of the soundboard. Figure 2(b) shows that the hammer-string interaction is bidirectional, since the hammer force depends on the string displacement [4]. On the other hand, there is no feedback from the radiator to the string. Feedback and coupling effects of the bridge and the soundboard are taken into account in the string block. Thus, at this point the model differs from the real piano: the two functions of the soundboard, namely the determination of decay times and the spectrum shaping, are put to separate parts of the model. This allows to treat radiation as a linear filtering operation.

# 3 The hammer

We first discuss physical aspects of hammer-string interaction, then concentrate on various modeling approaches and focus on accuracy and efficiency issues.

## 3.1 Hammer-string interaction

It is well known that the force-compression characteristics of a hammer felt is not described by the linear Hooke's law, i.e. the restoring force $f(x)$ is not simply proportional to the compression $x$. As a first approximation, a power law can be assumed:

$$y(x(t)) = f(x(t)) = kx(t)^p. \qquad (1)$$

It has been shown (see e.g. [13]) that Eq. (1) provides a qualitative description for real hammers with $p$-values ranging from $\sim 2$ to $\sim 4$. Due to this non-linearity, the tone spectrum varies dynamically with hammer velocity.

However, Eq. (1) is not fully satisfactory in that real piano hammers exhibit hysteretic behavior, i.e. a one-to-one law between compression and force does not adequately describe reality. Boutillon [14] proposed a model where non-constant values of the exponent $p$ account for different paths during loading and unloading of the felt; however this non-analytical model has no strong physical basis. A more general description of hysterisis was provided by Stulov [15]; the idea, coming from the general theory of mechanics of solids, is that the spring stiffness $k$ in Eq. (1) has to be replaced by a time-dependent operator. Thus the contact force for positive compressions takes the form

$$y(x(t)) = f(x(t)) = k[1 - h_r(t)] * [x(t)^p], \qquad (2)$$

where $h_r(t) = \frac{\epsilon}{\tau}e^{-t/\tau}$ is a *relaxation function* that controls the "memory" of the material. The Stulov model is successful in fitting experimental data where a hammer strikes a massive surface, and force, acceleration, displacement signal are recorded. However, recent research by Giordano and Mills [16] has investigated different experimental settings, where a hammer hits a vibrating string, and showed that the Stulov model is not able to fit the data collected from such an experiment. These results suggest the need for further investigations on alternative hammer models; one example is given by the collision model developed by Marhefka and Orin in [17].

## 3.2 Modeling approaches

As already mentioned in the Introduction, one way to account for hammer excitation is commuted synthesis [8]: in this approach the hammer is a linear filter and an excitation signal is simply provided to the string.

Alternatively, the models described in the previous section can be discretized and coupled to the string model, in order to provide a full physical description. It is easily seen that whichever method we use in order to translate the hammer equations (1) or (2) in discrete-time form, we obtain the structure depicted in Fig. 3(a), where **L** is a linear block, $u$ collects inputs to the hammer model and $y_E$ stands for the outputs. This results in an implicit system relating the $n$th sample of the force and the $n$th sample of the felt compression [18]. This implicit relationship can be made explicit by assuming that $y(n) \approx y(n-1)$, thus inserting a fictitious delay element in a delay-free path. Although this trick has been extensively used in the literature, it is a potential source of instability, as proved by Anderson and Spong [19]. Figure 3(b) shows that the insertion of a fictitious delay has severe consequences on the simulation of high-pitched notes at audio sampling rates.

A more rigorous approach to the problem is provided by the wave digital filter (WDF) theory; this can be generalized in order to provide a systematic methodology for modeling circuits (and mechanical systems) in which a non-linear element is present [20]. This approach was taken by Pedersini *et al.* [21], where a mechanical model of the hammer was connected to a WDF string model incorporating stiffness and distributed losses. Van Duyne and Smith [22] presented a distributed hammer model, connected to the waveguide by a scattering junction.

A rather general strategy for solving non-computable loops, named K method, has been recently proposed by Borin *et al.* [23]. We do not discuss details; suffice it to say that, whichever the discretization method, the hammer state $\boldsymbol{x}(n) := [x(n), \dot{x}(n)]^T$ can be written as

$$\boldsymbol{x}(n) = \boldsymbol{p}(n) + \boldsymbol{K}y(n), \qquad (3)$$

where $\boldsymbol{p}(n)$ is a computable vector (i.e. it is a linear combination of past values of $u$, $y$ and $x$) and $\boldsymbol{K}$ is the K matrix of the method. Substituting equation (3) in the non-linear contact force and applying the implicit function theorem we can find $f$ as a function of $\boldsymbol{p}$:

$$y = f\left(\boldsymbol{p} + \boldsymbol{K}y\right) \quad \overset{Kmeth.}{\longmapsto} \quad y = h(\boldsymbol{p}). \qquad (4)$$

Thus, instantaneous dependencies across the non-linearity are dropped. The function $h$ can be precomputed and stored in a look-up table for efficient implementation. The K method avoids artificial instabilities and allows to reproduce a reliable force signal (see dashed line in Fig. 3(b)), and more natural sounds. In fact, the bumps in the dashed line of Fig. 3(b) come from reflections from the string ends, while spikes in the solid line are not physical and are responsible for "buzzy" sounds. The K method was recently used by Avanzini and Rocchesso [24] for implementing the hysteretic hammer model by Marhefka and Orin [17]; high accuracy in the simulations was achieved at low computational costs.

Bank [25] presented a simpler, but less general method for avoiding artifacts caused by the fictitious delay. The idea is that instability can be avoided by increasing the sampling rate $F_s$. The discretized hammer model with inserted delay is stable when the variables change only a little in every temporal sampling interval, thus stability can be always maintained by choosing a sufficiently large $F_s$ (if the corresponding continuous-time system was stable). When $F_s \rightarrow \infty$, the discrete system will behave as the original differential equation. Increasing the sampling rate of the whole string model by a factor of two would double the computation time as well. Nevertheless, if only the hammer model operates at a double rate, the computational complexity is raised by a negligible amount. Therefore, in the proposed solution the string operates at normal, but the hammer runs at double $F_s$. For the downsampling, simple averaging, and for the upsampling, linear interpolation is used. The multi-rate hammer has been found to give well behaving force signals at a low computational cost.
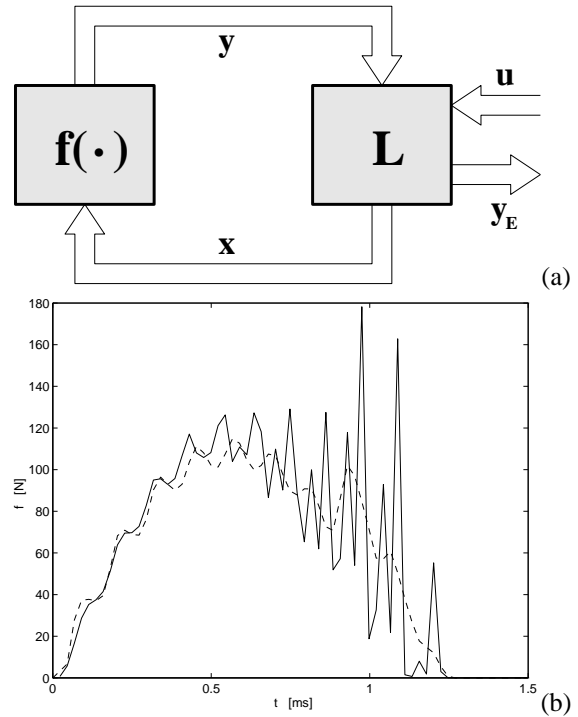


Figure 3: (a) Block scheme for a non-linear discrete-time hammer model. (b) Time evolution of $f$ with $F_s = 44.1$ [kHz], $v = 6 : 8$ [m/s] (fortissimo). Dashed line: K method solution. Solid line: insertion of fictitious delay element.

## 4 The string

Many different approaches have been presented in the literature for string modeling. Since we are considering techniques suitable for real-time applications, only the digital waveguide [6] is considered here in detail. This method is based on the time-domain solution of the one-dimensional wave equation: the velocity distribution $v(x, t)$ can be seen as the sum of two traveling waves:

$$v(x, t) = v^+(x - ct) + v^-(x + ct) \qquad (5)$$

where in this case $x$ denotes the spatial coordinate, $t$ is time, $c$ is the propagation speed, and $v^+$ and $v^-$ are the traveling wave components.

Spatial and time-domain sampling of Eq. (5) results in a simple delay-line representation. If the linearity and time-invariance of the string is assumed, all the distributed losses and dispersion can be consolidated to one end of the digital waveguide [6]. In the case of one polarization of a piano string, the system takes the form shown in Fig. 4, where $M$ is the length of the string in spatial sampling intervals, $M_{in}$ denotes the position of the force input, and $H_{r_v}(z)$ refers to the reflection filter. This structure is capable of generating a set of quasi-harmonic exponentially decaying sinusoids. The phase response of $H_{r_v}(z)$, together with the total delay line length are responsible for controlling the frequencies of the partials. The decay times of the partials are determined by the magnitude response of $H_{r_v}(z)$ and the total length of the delay line.
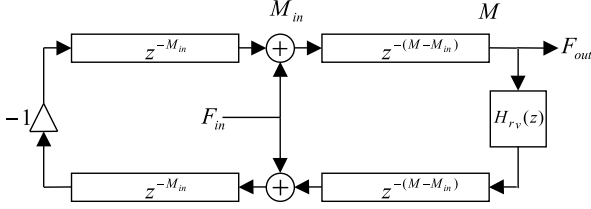
Figure 4: Digital waveguide model of a string with one polarization.

For creating realistic sounds, accurate design of the reflection filter plays a key role. To simplify the design, it is usually factored into three parts: $H_{r_v} = -H_l H_d H_{fd}$, where $H_l$ accounts for the losses, $H_d$ for the dispersion, and $H_{fd}$ for fine-tuning the fundamental frequency. Using allpass filters $H_d$ for simulating dispersion ensures that the decay times of the partials are controlled by the loss filter $H_l$ only. The slight phase difference caused by the loss filter is negligible compared to the phase response of the dispersion filter. This way, the loss filter and dispersion filter can be treated as orthogonal with respect to design, as we do in the next two sections. Fine tuning of the string is needed because only an integer phase delay can be implemented with delay lines and this provides a too rough discretization of the allowed fundamental frequencies. Fractional delay can be incorporated in the dispersion filter design, of alternatively a separate fractional delay filter $H_{fd}$ can be used in series with the delay line. Jaffe and Smith [26, 5] suggested to use a first-order allpass filter for this purpose. Välimäki *et al.* [27] proposed an implementation based on low-order Lagrange interpolation filters. Välimäki [28] and Laakso *et al.* [29] provided exhaustive overviews on this topic.

## 4.1   Loss filter design

First, the partial envelopes of the measured note has to be calculated. This can be done by the Short Time Fourier Transform [27] or by heterodyne filtering [30]. A robust way of calculating decay times is fitting a line by linear regression on the logarithm of the amplitude envelopes [27]. The magnitude specification $g_k$ for the loss filter can be computed as follows:

$$g_k = \left| H_l \left( e^{j \frac{2\pi f_k}{F_s}} \right) \right| = e^{-\frac{k}{f_k \tau_k}} \tag{6}$$

where $f_k$ and $\tau_k$ are the frequency and the decay time of the $k^{\text{th}}$ partial, and $F_s$ is the sampling rate. Fitting a filter to the $g_k$ coefficients is not trivial, since the error in the decay times is a non-linear function of the filter magnitude error. If the magnitude response oversteps unity, the digital waveguide loop becomes unstable. To overcome this problem, Välimäki *et al.* [27, 30] suggested the use of the one-pole loop filter, whose transfer function is:

$$H_{1p}(z) = g \frac{1 + a_1}{1 + a_1 z^{-1}} \tag{7}$$

The advantage of this filter is that stability constraints for the waveguide loop are relatively simple, namely $a_1 < 0$ and $0 < g < 1$. As for the design, Välimäki *et al.* [27, 30] used a simple algorithm for minimizing the magnitude error in the mean squares sense. However, the overall decay time of the synthesized tone did not always coincide with the original one. Erkut *et al.* [31] suggested an iterative optimization algorithm to overcome this problem.

As a general solution for loss filter design, Bank [11] suggested to minimize the approximation error in the decay time domain. This assures that the overall decay time of the note is ensured together with the stability of the feedback loop. Moreover, optimization with respect to decay times is perceptually more meaningful than minimizing the error of the filter magnitude response. The methods described hereafter are all based on this idea.

The approximate analytical formulas for the decay times $\tau_k$ of a digital waveguide with a one-pole filter (7) were given by Bank [11]:

$$\tau_k \approx \frac{1}{c_1 + c_3 \vartheta_k^2} \tag{8}$$

which is the same as for a string with the simplest frequency dependent losses; $c_1$ and $c_3$ correspond to the first and third order time derivatives of the wave equation:

$$\begin{aligned} c_1 &= f_0(1 - g) \\ c_3 &= -f_0 \frac{a_1}{2(a_1 + 1)^2} \end{aligned} \tag{9}$$

where $f_0$ is the fundamental frequency and $\vartheta_k$ is the digital frequency of the $k^{\text{th}}$ partial. Equation (8) shows that the decay rate $\sigma_k = 1/\tau_k$ is a second order polynomial of frequency $\vartheta_k$ with even order terms. This simplifies the filter design, since $c_1$ and $c_3$ are easily determined by polynomial regression. A weighting function has to be used to minimize the error with respec $\tau_k$, and not to decay rates. From the $c_1$, $c_3$ coefficients the parameters of the one-pole loop filter are easily computed via Eq. (8).

For the precise rendering of the partial envelopes, higher-order filters have to be used. However, computing analytical formulas for the decay times with high-order filters is a difficult task. A two-step procedure was suggested by Erkut [32]; in this case, a high-order polynomial is fit to the decay rates, which contains even order terms only. Then, a magnitude specification is calculated from the decay rate curve defined by the polynomial and this magnitude response is used as a specification for minimum-phase filter design. Another approach was proposed by Bank [11], who suggested the transformation of the specification. As the goal is to match the decay times, the magnitude specification $g_k$ is transformed in a form $g_{tr}$ which approximates $\tau_k$, and a transformed filter $H_{tr}$ is designed for the new specification by minimizing $e_{LS}$:

$$e_{LS} = \sum_{k=1}^{K} w_k \left( H_{tr}(e^{j\vartheta_k}) - g_{k,tr} \right)^2, \quad g_{k,tr} = \frac{1}{1 - g_k} \tag{10}$$

The loss filter $H_l(z)$ is then computed by the inverse transform $H_l = 1 - 1/H_{tr}$. Both of these techniques

for high-order loss filter design have found to be robust in practice. Comparing them is left for future work.

Borin *et al.* [10] have used a different approach for modeling the decay time variations of the partials. In their implementation, second order FIR filters are used as loss filters. These are responsible for the general decay of the note. The small variations of the decay times are modeled by connecting all the notes to a same termination, which is a complex filter with a high number of resonances. This also enables the simulation of the pedal effect, since now all the strings are coupled to each other (see Sec. 4.3). An advantage of this method compared to high-order loop filters is the smaller computational complexity. On the other hand, the partial envelopes of the different notes cannot be controlled independently.

Although optimizing the loss filter with respect to decay times has been found to give perceptually adequate results, we remark that the loss filter design can be helped via perceptual studies. The audibility of the decay-time variations for the one-pole loss filter was studied by Tolonen and Järveläinen [33]. The study states that relatively large deviations (between $-25\%$ and $+40\%$) in the overall decay time of the note are not perceived by listeners. Unfortunately, the results of the paper are not directly applicable for the design of high-order loss filters.

## 4.2 Dispersion simulation

Dispersion is due to stiffness, that cause piano strings to behave differently from an ideal string obeying the wave equation. If the dispersive correction term is small, its first order effect is to increase the wave propagation speed $c(f)$ with frequency. This phenomenon cause the string partials to become inharmonic: if the string parameters are known, then the frequency of the $k^{th}$ stretched partial can be computed as

$$f_k = k f_0 \sqrt{1 + Bk^2}, \qquad (11)$$

where the value of the inharmonicity coefficient $B$ depends on string parameters.

The problem of simulating dispersion in a waveguide structure is equivalent to designing a filter $H_d(f)$ with flat magnitude response and a non-linear phase response accounting for the frequency dependent wave velocity $c(f)$ [6]. Van Duyne and Smith [34] proposed a very efficient method for simulating dispersion by cascading equal first-order allpass filters in the waveguide loop; however, the constraint of using equal first order sections is too severe, and does not allow accurate tuning of inharmonicity.

Rocchesso and Scalcon [35] proposed a design method based on [36]. Starting from the desired phase response, $l$ points $\{f_k\}_{k=1...l}$ are chosen on the frequency axis, corresponding to the points where the string partials should lie; the filter order is chosen to be $n < l$. For each partial $k$ the method computes the quantities

$$\beta_k = -\frac{1}{2} \left( \phi_{pre}(f_k) + 2n\pi f_k \right), \qquad (12)$$
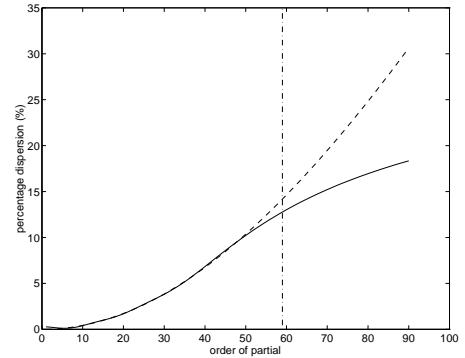


(a)
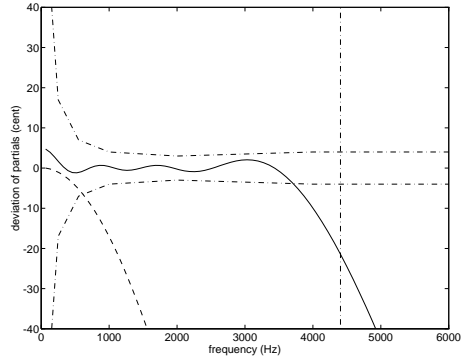


(b)

Figure 5: Synthesis of dispersive filters for the $C_2$ string with three sixth-order filters; (a) percentage dispersion vs. partial numbers and (b) deviation of partials.

where $\phi_{pre}(f)$ is the prescribed allpass response. Then it find the filter coefficients $a_j$ by solving the system

$$\sum_{j=1}^{n} a_j \sin(\beta_k + 2j\pi f_k) = -\sin(\beta_k), \qquad k = 1 \ldots l \qquad (13)$$

A Least-Squared Equation Error (*LSEE*) is used to solve the overdetermined system (13). It was showed in [35] that several tens of partials can be correctly positioned for any piano key, with the allpass filter order not exceeding 20. Moreover, fine tuning of the string is automatically taken into account in the design. Figure 5 plots results obtained with three sixth-order filters. A vertical line shows where the approximation in the LSEE method ends, while the two bounds in Fig. 5(b) indicate the frequency JND (Just Noticeable Difference). The steep dashed line is the partial distribution in a non-dispersive string.

Since the computational load of $H_d$ is heavy, it is important to find criteria that allows to optimize accuracy and order of the filters with respect to human perception. Rocchesso and Scalcon [37] studied the dependence of the bandwith of perceived inharmonicity on the fundamental frequency, by performing listening tests with decaying piano tones; such a bandwidth is seen to increase almost linearly on a logarithmic pitch scale. Järveläinen *et al.* [38] also found that inharmonicity is more easily perceived at low frequencies, even when the $B$ coefficient

for bass tones is lower than for treble tones. This is probably due to the fact that beats are used by listeners as a cue for inharmonicity, and even low $B$'s produce enough mistuning in higher partials of low tones. This findings can help in the allpass filter design procedures, though there is still a number of issues that need further investigations.

As high-order dispersion filters are needed for modeling low notes, they increase the computational complexity significantly. Bank [11] proposed a multi-rate approach to overcome this problem. Since the lowest tones do not contain significant energy in the high frequency region anyway, it is worthwhile to run the lowest two or three octaves of the piano at the half of the sampling rate of the model. This will reduce the required computation in two ways: one is that the whole digital waveguide loop has to be computed for every second time instant only. The other is the dispersion filter gets simpler, since the total length of the digital waveguide diminishes by a factor of two in terms of delay elements. The outputs of the low notes are summed before upsampling, therefore only one interpolation filter is required.

## 4.3 Coupled piano strings

Coupling between strings occurs at two different levels: first of all, two or three slightly mistuned strings are sounding together when a single piano key is depressed (except for the lowest octave) and a complicated modulation of the amplitudes is brought about. This results in beating and two-stage decay, the first referring to an amplitude modulation overlaid on the overall tone decay, and the latter meaning that tone decay is not exponential and is faster in the beginning. These phenomena were studied by Weinreich already in 1977 [39]. At a second level, the presence of the bridge and the action of the soundboard is known to originate important coupling effects even between different tones. In fact, the bridge–soundboard system connects the strings together and acts as a distributed driving-point impedance for string terminations.

The simplest way for modeling beating and two-stage decay is to use two digital waveguides in parallel for a single note. When their pitches are different, beating, when their decay times are different, two-stage decay will appear in the sound. Karjalainen *et al.* [40] suggested the use of real coupling coefficients. Nevertheless, the envelopes of specific partials cannot be controlled individually by this model, they will have similar behavior.

Another approach, taken by Smith [41], couples two strings to the same termination and lumps all the losses to the bridge impedance. This comes from the assumption that all the losses come from the bridge, which is a rough approximation. One advantage is that only one loss filter is needed, whose transfer function can be determined from the decay times of the partials. The drawback is that decay times and mode coupling are not independent.

Aramaki *et al.* [42] presented a model of coupled waveguides with four filters. Two of them accounted for the losses and dispersion of the strings and two for the coupling. By this increased degree of freedom, the model

was able to accurately simulate the sound of two coupled piano strings with one-polarization. However, the model was implemented in the frequency domain, which makes it unrealizable for real-time applications. For time domain implementation, high-order coupling filters should be designed, and no such filter design methods exist which guarantee the stability of such a coupled system.

In these proceedings, Bank [43] presents a different model beating and two stage decay, based on a multi-rate resonator bank. In this approach, second order resonators are connected to the basic string model in parallel, instead of using a second waveguide. The resonator bank is implemented by the multi-rate approach, resulting in significantly lower computational costs, compared to the methods mentioned earlier. The parameter estimation gets simpler, since there is no need for coupling filter design. Stability problems of a coupled system are also avoided.

Modeling the coupling between strings of different tones is essential when the sustain pedal effect has to be simulated. Garnett [7] and Borin *et al.* [10] suggested to connect the strings to the same lumped terminating impedance. The impedance is modeled by a filter with a high number of peaks, for that, the use of feedback delay networks [44] is a good alternative. Although in real pianos the bridge connects the string as a distributed termination, thus coupling different strings in different ways, the simple model of Borin *et al.* was able to produce realistic sustain pedal effect [45].

# 5 The radiation problem

The soundboard radiates and filters the velocity waves that reach the bridge, and radiation patterns are essential for describing the "presence" of a piano in a musical context. Modeling the soundboard as a linear post-processing stage is intrinsically a weak approach, since in a real piano it also accounts for coupling between strings, and affects the decay times of the partials. However, as already stated in Sec. 2, our modeling strategy keeps the *radiation* properties of the soundboard as separated from its *impedance* properties. The latter are incorporated in the string model, and have already been addressed in Sec. 4.1 and 4.3; here we concentrate on radiation.

The most efficient approach for modeling the radiation effect of the soundboard is the commuted piano model of Van Duyne and Smith [8]. There, the soundboard is commuted with the string and the hammer models. In order to do this, the whole system must be assumed to be linear, including the hammer. The advantage of the method is that the soundboard is not implemented as a high-order filter, but as a wavetable, whose content is fed to the string. Since this approach cannot be used with non-linear hammers, we do not consider it in detail.

Giordano [46] presented a finite difference piano soundboard model. A similar model was developed by Bazzi and Rocchesso [47]. This approach is physically meaningful and allows simulation of many features of the soundboard, e.g. the effect of ribs. On the other hand,
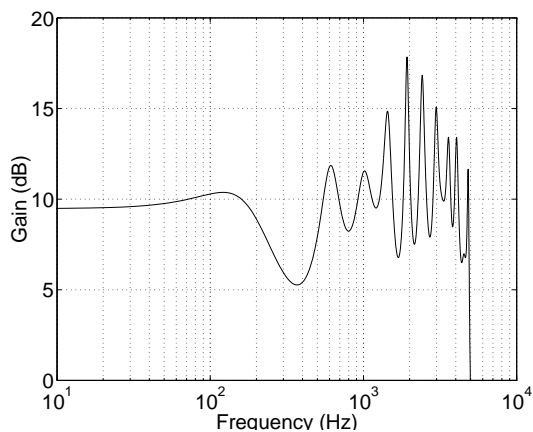
Figure 6: Soundboard magnitude response obtained with the CAP method.

solving the difference equations takes too much computations, and is not suitable for real-time applications.

Multidimensional Digital Waveguide Networks (N-D DWN), first proposed in a particular case known as Waveguide Mesh [48], can describe a wide variety of propagation phenomena, and have straightforward translation to parallel algorithms. These structures were recently adopted for modeling stringed instrument bodies [49]. However, while they are successful in simulating vibrations in elastic media, they do not naturally fit in models of stiff bodies (as a piano soundboard). At the moment, we are not aware of a successful description of the soundboard by means of a DWN.

Fontana [50] obtained accurate radiation spectral responses by extracting common spectral features from pre-recorded piano samples in which the contribution of the soundboard is present. The Common Acoustical Poles (CAP) method by Haneda *et al.* [51] allows to calculate the coefficients of a FIR filter with a desired order, that matches these common spectral features. The algorithm must be fed using properly pre-processed data in order to avoid any possible detection of common zeros or singularities by the algorithm that would translate into huge peaks or dips. Figure 6 shows the magnitude response of such a model, consisting of a $126^{th}$-order FIR filter computed with the CAP method. The algorithm was fed using a collection of high-quality, anechoic piano samples, and the response was limited to 5 kHz. Although the impulse response of this filter is not comparable with that (much longer, and position-dependent) of a real soundboard, its performance suggests that carefully designed linear post-processors may be used in connection with proper reverberation stages and loudspeakers.

A simple and efficient radiation model was presented by Garnett [7]. The waveguide strings were connected to the same termination and the soundboard was simulated by connecting six additional waveguides to the common termination. Each of these waveguides incorporated a lowpass filter with a large damping factor. This can be seen as a predecessor of using feedback delay networks for soundboard simulation. Feedback delay networks have been proven to be efficient in simulating room reverberation, since they are able to produce high modal density at a low computational cost. For an overview, see the work of Rocchesso and Smith [44]. Bank [11] applied feedback delay networks with shaping filters for the simulation of piano soundboards. The shaping filters were set in such a way that the system matched the overall magnitude response of a real piano soundboard. A drawback of the method is that the modal density and the quality factors of the modes are not fully under control. Tuning the parameters of the feedback delay network by hand requires a significant amount of work and not always results in a satisfactory sound. The method has proven to be rather applicable for high piano notes, where simulating the attack noise (the knock) of the tone is the most important issue. However, no satisfactory results are obtained for the treble and bass registers.

# 6   Conclusions

We have reviewed the main stages in the development of a physical model for the piano, addressing computational aspects in detail. We have showed that computational loads are due to both the presence of non-linearities and the need of high-order filtering elements.

Various approaches have been discussed for dealing with non-linear equations in the excitation block; we have pointed out that inaccuracies at this stage can lead to severe instability problems. Investigation of alternative non-linear hammer models is an appealing topic for future research. However, we emphasize that in our opinion using more accurate models would probably not increase the overall sound quality substantially.

Several filter design techniques have been reviewed for the accurate tuning of the resonating waveguide block. Especially the dispersion filter has been shown to require high orders for accurate simulation of inharmonicity. This is why perceptual studies can be helpful in optimizing the design and reducing computational loads.

We do believe that radiation modeling is at present the most urgent topic to be addressed in order to step toward high sound quality. None of the approaches presented in Sec. 5 has so far provided satisfactory results; on the other hand, when experimenting with large (e.g. 2000 tap) FIR filters good results can be achieved. This suggests that there is still room for improvements, even when radiation is modeled as a linear filtering operation. The development of parameter estimation for feedback delay networks could be of significant help in deriving a satisfactory radiation model at low computational costs.

# Acknowledgements

# References

[1] G. De Poli, "A Tutorial on Digital Sound Synthesis Techniques," in *The Music Machine*, C. Roads, Ed., pp. 429–447. MIT Press, 1991.

[2] J. O. Smith III, "Viewpoints on the History of Digital Synthesis," in *Proc. Int. Computer Music Conf. (ICMC'91)*, Montreal, Oct. 1991, pp. 1–10.

[3] K. Tadamura and E. Nakamae, "Synchronizing Computer Graphics Animation and Audio," *IEEE Multimedia*, vol. 5, no. 4, pp. 63–73, Oct. 1998.

[4] N. H. Fletcher and T. D. Rossing, *The Physics of Musical Instruments*, Springer-Verlag, New York, 1991.

[5] J. O. Smith III, *Techniques for Digital Filter Design and System Identification with Application to the Violin*, Ph.D. thesis, Stanford University, California, USA, June 1983.

[6] J. O. Smith III, "Principles of Digital Waveguide Models of Musical Instruments," in *Applications of DSP to Audio and Acoustics*, M. Kahrs and K. Brandenburg, Eds., pp. 417–466. Kluwer Academic Publishers, 1998.

[7] G. E. Garnett, "Modeling Piano Sound using Digital Waveguide Filtering Techniques," in *Proc. Int. Computer Music Conf. (ICMC'87)*, Urbana, Illinois, USA, 1987, pp. 89–95.

[8] S. A. Van Duyne and J. O. Smith III, "Developments for the commuted piano," in *Proc. Int. Computer Music Conf. (ICMC'95)*, Banff, Sept. 1995, pp. 335–343.

[9] G. Borin, G. De Poli, and A. Sarti, "A Modular Approach to Excitator-Resonator Interaction in Physical Models Synthesis," in *Proc. Int. Computer Music Conf. (ICMC'90)*, Columbus, OH, 1990, pp. 46–50.

[10] G. Borin, D. Rocchesso, and F. Scalcon, "A Physical Piano Model for Music Performance," in *Proc. Int. Computer Music Conf. (ICMC'97)*, 1997, pp. 350–353.

[11] B. Bank, "Physics-Based Sound Synthesis of the Piano," M.S. thesis, Budapest University of Techology and Economics, Hungary, May 2000, Published as Report 54 of HUT Laboratory of Acoustics and Audio Signal Processing, available at http://www.mit.bme.hu/~bank.

[12] F. Fontana, "Riproduzione della Dinamica del Pianoforte," Internal report GEM 6, Generalmusic S.p.A., July 2000.

[13] D. E. Hall, "Piano String Excitation VI: Nonlinear Modeling," *J. Acoust. Soc. Am.*, vol. 92, pp. 95–105, July 1992.

[14] X. Boutillon, "Model for Piano hammers: Experimental Determination and Digital Simulation," *J. Acoust. Soc. Am.*, vol. 83, pp. 746–754, 1988.

[15] A. Stulov, "Hysteretic Model of the Grand Piano Hammer Felt," *J. Acoust. Soc. Am.*, vol. 97, no. 4, pp. 2577–2585, Apr. 1995.

[16] N. Giordano and J. P. Mills, "Hysteretical Behavior of Piano Hammers," in *Proc. Int. Symp. Mus. Acoust. (ISMA'01)*, Perugia, Sept. 2001, pp. 237–241.

[17] D. W. Marhefka and D. E. Orin, "A Compliant Contact Model with Nonlinear Damping for Simulation of Robotic Systems," *IEEE Trans. Systems, Man and Cybernetics-Part A*, vol. 29, no. 6, pp. 566–572, Nov. 1999.

[18] G. Borin and G. De Poli, "A Hysteretic Hammer-String Interaction Model for Physical Model Synthesis," in *Proc. Nordic Acoustical Meeting*, Helsinki, June 1996, pp. 399–406.

[19] R. J. Anderson and M. W. Spong, "Bilateral Control of Teleoperators with Time Delay," *IEEE Trans. Automat. Contr.*, vol. 34, no. 5, pp. 494–501, May 1989.

[20] A. Sarti and G. De Poli, "Toward Nonlinear Wave Digital Filters," *IEEE Trans. Sig. Process.*, vol. 47, no. 6, pp. 1654–1668, June 1999.

[21] F. Pedersini, A. Sarti, and S. Tubaro, "Block-Wise Physical Model Synthesis for Musical Acoustics," *IEEE Electronics Letters*, vol. 35, no. 17, pp. 1418–1419, Aug. 1999.

[22] S. A. Van Duyne, J. R. Pierce, and J. O. Smith III, "Traveling Wave Implementation of a Lossless Mode-Coupling Filter and the Wave Digital Hammer," in *Proc. Int. Computer Music Conf. (ICMC'94)*, Århus, Denmark, Sept. 1994, pp. 411–418.

[23] G. Borin, G. De Poli, and D. Rocchesso, "Elimination of Delay-free Loops in Discrete-Time Models of Nonlinear Acoustic Systems," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 597–606, Sep. 2000.

[24] F. Avanzini and D. Rocchesso, "Controlling Material Properties in Physical Models of Sounding Objects," in *Proc. Int. Computer Music Conf. (ICMC'01)*, Cuba, Sept. 2001.

[25] B. Bank, "Nonlinear Interaction in the Digital Waveguide with the Application to Piano Sound Synthesis," in *Proc. Int. Computer Music Conf. (ICMC'00)*, Berlin, Germany, 2000.

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

[27] V. Välimäki, J. Huopaniemi, M. Karjalainen, and Z. Jánosy, "Physical Modeling of Plucked String Instruments with Application to Real-Time Sound Synthesis," *J. Aud. Eng. Soc.*, vol. 44, no. 5, pp. 331–353, May 1996.

[28] V. Välimäki, *Discrete-Time Modeling of Acoustic Tubes Using Fractional Delay Filters*, Ph.D. thesis, Helsinki Univ. of Technology, Lab. of Acoustics and Audio Signal Process., 1995, available at www.acoustics.hut.fi/publications/theses.html.

[29] T. I. Laakso, V. Välimäki, M. Karjalainen, and U. K. Laine, "Splitting the Unit Delay Tools for Fractional Delay Filter Design," *IEEE Sig. Process. Magazine*, vol. 13, no. 1, pp. 30–60, Jan. 1996.

[30] V. Välimäki and T. Tolonen, "Development and Calibration of a Guitar Synthesizer," *J. Aud. Eng. Soc.*, vol. 46, no. 9, pp. 766–778, Sept. 1998.

[31] C. Erkut, V. Välimäki, M. Karjalainen, and M. Laurson, "Extraction of Physical and Expressive Parameters for Model-Based Sound Synthesis of the Classical Guitar," in *Proc. 108$^{\text{th}}$ AES conv., Preprint No. 5114*, Paris, France, Feb. 2000.

[32] C. Erkut, "Loop Filter Design Techniques for Virtual String Instruments," in *Proc. Int. Symp. Mus. Acoust. (ISMA'01)*, Perugia, Sept. 2001, pp. 259–262.

[33] T. Tolonen and H. Järveläinen, "Perceptual Study of Decay Parameters in Plucked String Synthesis," in *Proc. AES 109$^{\text{th}}$ Int Conv., Preprint No. 5205*, 2000.

[34] S. A. Van Duyne and J. O. Smith III, "Simplified Approach to Modeling Dispersion Caused by Stiffness in Strings and Plates," in *Proc. Int. Computer Music Conf. (ICMC'94)*, Århus, Denmark, Sept. 1994, pp. 407–410.

[35] D. Rocchesso and F. Scalcon, "Accurate Dispersion Simulation for Piano Strings," in *Proc. Nordic Acoustical Meeting*, Helsinki, June 1996, pp. 407–414.

[36] M. Lang and T.I. Laasko, "Simple and Robust Method for the Design of Allpass Filters," *IEEE Trans. Circuits Systems*, vol. 41, no. 1, pp. 40–48, 1994.

[37] D. Rocchesso and F. Scalcon, "Bandwidth of Perceived Inharmonicity for Physical Modeling of Dispersive Strings," *IEEE Trans. Speech Audio Process.*, vol. 7, no. 5, pp. 597–601, Sept. 1999.

[38] H. Järveläinen, V. Välimäki, and M. Karjalainen, "Audibility of Inharmonicity in String Instruments Sounds, and Implications to Digital Sound Synthesis," in *Proc. Int. Computer Music Conf. (ICMC'99)*, Bejing, 1999, pp. 220–223.

[39] G. Weinreich, "Coupled Piano Strings," *J. Acoust. Soc. Am.*, vol. 62, no. 6, pp. 1474–1484, Dec. 1977.

[40] M. Karjalainen, V. Vlimki, and T. Tolonen, "Plucked-String Models: from Karplus-Strong Algorithm to Digital Waveguides and Beyond," *Computer Music J.*, vol. 22, no. 3, pp. 17–32, 1998.

[41] J. O. Smith III, "Efficient Synthesis of Stringed Musical Instruments," in *Proc. Int. Computer Music Conf.*, Tokyo, Japan, Sept. 1993, pp. 64–71.

[42] M. Aramaki, J. Bensa, L. Daudet, Ph. Guillemain, and R. Kronland-Martinet, "Resynthesis of Coupled Piano String Vibrations Based on Physical Modeling," *to be published in the Journal of New Music Research*.

[43] B. Bank, "Accurate and Efficient Modeling of Beating and Two-Stage Decay for String Instrument Synthesis," in *Proc. MOSART Workshop on Curr. Res. Dir. in Computer Music*, Barcelona, Spain, 2001, elsewhere in these proceedings.

[44] D. Rocchesso and J. O. Smith III, "Circulant and Elliptic Feedback Delay Networks for Artificial Reverbiration," *IEEE Trans. Speech Audio Process.*, vol. 5, no. 1, pp. 51–63, Jan. 1997.

[45] G. De Poli, F Campetella, and G. Borin, "Pedal Resonance Effect Simulation Device for Digital Pianos," United States Patent 5,744,743, Apr. 28, 1998, (Appl. No. 618379, filed: March 19, 1996).

[46] N. Giordano, "Simple Model of a Piano Soundboard," *J. Acoust. Soc. Am.*, vol. 102, no. 2, pp. 1159–1168, Aug. 1997.

[47] B. Bazzi and D. Rocchesso, "Numerical Investigation of the Acoustic Properties of Piano Soundboars," in *Proc. XIII Colloquium Mus. Inf. (CIM'00)*, L'Aquila, Sept. 2000, pp. 39–42.

[48] S. 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)*, New Paltz (NY), Oct. 1993.

[49] P. Huang and S. Serafin, "Modeling High Frequencies Modes of Complex Resonators Using a Waveguide Mesh," in *Proc. COST-G6 Conf. Digital Audio Effects*.

[50] F. Fontana, "Un Modello per la Rimozione del Colore e la Ricolorazione in Tempo Reale di un Set di Campioni di Pianoforte," Internal report GEM 4, Generalmusic S.p.A., Feb. 2000.

[51] Y. Haneda, S. Makino, and Y. Kaneda, "Common Acoustical Pole and Zero Modeling of Room Transfer Functions," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 2, pp. 320–328, Apr. 1994.