SIGNAL- AND PHYSICS-BASED SOUND SYNTHESIS OF MUSICAL INSTRUMENTS

Balázs BANK*, János MÁRKUS*, Attila NAGY** and László SUJBERT*

*Department of Measurement and Information Systems
Budapest University of Technology and Economics
H–1521 Budapest, Hungary
e-mail: bank@mit.bme.hu, markus@mit.bme.hu, sujbert@mit.bme.hu
**Department of Telecommunications
Budapest University of Technology and Economics
H–1521 Budapest, Hungary
e-mail: nagyab@hit.bme.hu

Abstract

In this paper signal-based and physics-based sound synthesis methods are described, with a particular emphasis on our own results achieved in the recent years. The applications of these methods are given for the case of organ, piano, and violin synthesis. The two techniques are compared based on these case studies, showing that in some cases the physics-based, in other cases the signal-based realization is more advantageous. As a theoretical result, we show that the two methods can be equivalent under special circumstances.

Keywords: digital signal processing, sound synthesis, musical acoustics, signal modelling, physical modelling, organ, piano, violin.

1. Introduction

Musicians and music students – especially those playing the organ, the piano or other large instruments – require to have small size, economic and light musical instruments for portable, stage or home applications. Composers would like to try all kinds of instruments they otherwise do not play for searching new ways of expression. Thus, models of traditional instruments are needed to satisfy these requirements. Naturally, the sound quality of these artificial instruments needs to be comparable to that of the original ones. By modelling traditional instruments (like guitar, piano, organ, strings, winds, brass, etc.) and modifying the model parameters, novel, never-heard sounds can be generated. In addition, with more insight and better description of the physical operation of these instruments, new and efficient algorithms can be developed from which other fields of digital signal processing can benefit.

Sound synthesis methods can be classified in many ways. Here we divide them into three groups, by unifying two groups of the classifications found in [1].

The first group is the family of *abstract methods*. These are different algorithms which can easily generate synthetic sounds. Methods like frequency

modulation [2] and waveshaping [3, 4] belong to this category. Modelling real
instruments with these methods is fairly complicated as the relationship between
the parameters of the technique and those of the real instruments cannot be easily
formulated. Thus, these methods are beyond the scope of this paper.

The second group (*signal modelling*) is the one which models the *sound* of
musical instruments. In this case, the input to the model is only the waveform or a set
of waveforms generated by the instrument and the physics of the sound generation
mechanism is not examined in detail. Synthesis methods like sampling [5] and
SMS (Spectral Modelling Synthesis) [6] belong to this category. The corresponding
groups in the taxonomy of [1] are *processing of pre-recorded samples* and *spectral
models*.

The third group (*physical modelling*) is the one which instead of reproducing
a specific sound of an instrument, tries to model the instrument's physical behaviour
itself. Usually, the physical system (such as a string on an instrument or the skin
of a drum) can be described with a set of partial differential equations and transfer
functions. Given the excitation of the instrument (such as bowing the string or
hitting the drum), the difference equations can be solved (or the general solution
can be applied for the given input), and the output of the model is expected to be
close to the output of the real instrument. One well-known method in this category
is the digital waveguide synthesis [7] which efficiently models the vibration of a
one-dimensional string, based on the solution of the wave-equation.

In this paper, the signal- and physical-model based synthesis methods are
examined, based on our own results achieved in the last years. In Section 2 an effi-
cient signal model based synthesis method is introduced and applied for modelling
the sound of organ pipes. Then Section 3 describes an extended digital-waveguide
based physical model with the application of modelling the sound of the piano and
the violin. Finally, in Section 4, the equivalence of the two methods for a given ex-
citation is proven, and detailed comparison is given from the viewpoint of efficiency
and applicability. The results are summarized in Section 5.

2. Signal Modelling

Nowadays, the most commonly used signal-model based synthesis method is the
sampling method (sometimes referred as PCM – Pulse Code Modulation). This
method samples the sound of the instrument to be modelled, stores the samples in
a digital memory and plays it back when required. To reduce the required memory
for a waveform, usually the quasi-steady state of the sound is stored as one period,
and this period is repeated continuously at playback. To be even more effective,
usually not all the notes are sampled (e.g. all the 88 keys of a piano), but only a few,
and the missing waveforms are generated by resampling the stored ones.

It can be readily deducted from the above description that the sampling syn-
thesis technique has some limitations. One limitation is the lack of controllability.
As the method simply plays back the wavetables, the musician has only limited

tools to modify the characteristics of the sound (such as linear filters and amplitude envelopes). Other limitation is the absence of random effects. Most instrument (especially bowed string instruments and wind instruments) produce different transients at the start of the sound and random effects exist also in the stationary state (e.g. the so-called wind-noise in the case of wind instruments).

Thus, a signal model has to take into account all these effects. In the following, first the organ and its characteristics from a signal modelling viewpoint are described. Then a conceptual signal model and its application to the sound synthesis of the organ pipe is introduced which is flexible enough to model all the required parameters.

2.1. The Sound Characteristics of the Organ

The pipe-organ is one of the largest musical instruments. A small, efficient and high fidelity instrument substituting the church organ is long awaited by the organ players and students. Accordingly, the organ is among the most intensively studied instruments.

The sound generators of the organ are the flue and the reed pipes. As in a real organ flue pipes are dominant (small organs do not even have reed pipes), only the most important properties of the sound generated by the flue pipes are examined in the following.

It is well known that a significant and also easy-to-measure part of a musical instrument's sound is the stationary spectrum. Accordingly, the organ stops have also different characters, and the spectrum is strongly dependent on the pipes' physical parameters [8]. In addition, the way the sound builds up and tails off (the attack and decay transients of the sound) and the modulations on the harmonics, or other quasi-steady processes are important parts of a musical sound, too. Examinations prove that without the attack and decay transients some instruments cannot be identified [9], and in some other cases only the specific modulations of an instrument on a sine wave are enough to recognize the instrument itself [10]. Hence, a good synthesis has to take into account both the transient and the quasi-steady processes.

Another property of some musical instruments is the effect of the ambience of the sound source. The organ normally sounds in a church or in a hall, far away from the listeners. Closer to the pipes (without reverberation) the organ sounds unfamiliar [11]. Another external effect is the sometimes observable coupling mechanism of two pipes [12]. The localization of the sound-sources (which originates from the different positions of the pipes) falls also under this category [13].

2.2. Model Structure

The main concept of the proposed synthesis method is the periodic signal model which is based on the Fourier-expansion of the periodic signals. Such a generator

can generate a band-limited periodic signal, consisting of N complex components [14]. In sound synthesis it realizes directly the discrete spectrum components (the partials) of the instrument, and it is usually referred to as additive synthesis [5]. Studying the organ sound, the attack and decay transients can be easily realized by modifying the amplitude-envelope of these partials at the beginning and at the end of the sound.

For the organ pipes, the most important quasi-steady sound is the wind-noise. In some stops, this is the component which characterizes the pipe, thus it needs to be modelled. It can be seen in *Fig. 3*, that the noise is a wide-band component of the sound, with a typical spectral shape. To integrate it into the signal model, the periodic generator can be completed with a noise-generator. Naturally, during the transients the envelope of the noise has to be changed as well.

The applied periodic signal model for sound synthesis is displayed in *Fig.1*. The periodic signal generator has two main parameters – the fundamental frequency and the volume – and each harmonic component has further parameters, the relative magnitude and the phase. The noise generator produces filtered white noise, which is added to the magnitude-modified outputs of the periodic generator. At the end the summarized output is modified by the external effects discussed above, such as reverberation.
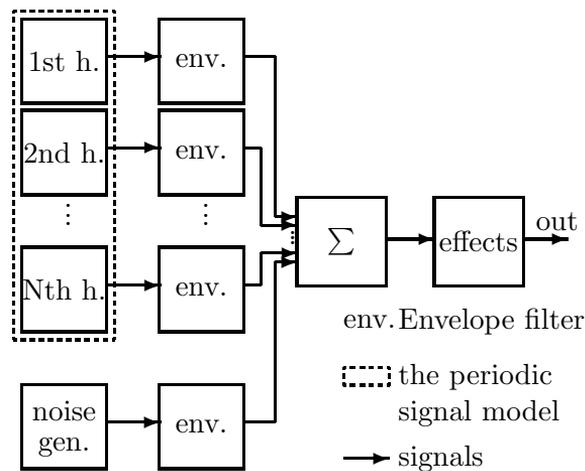
Fig. 1. The integrated signal model.

2.3. *Parameter Estimation*

In order to determine the correct parameters of the signal model, real pipe sounds were recorded and processed off-line with MATLAB, using the analysis process displayed in *Fig. 2*.
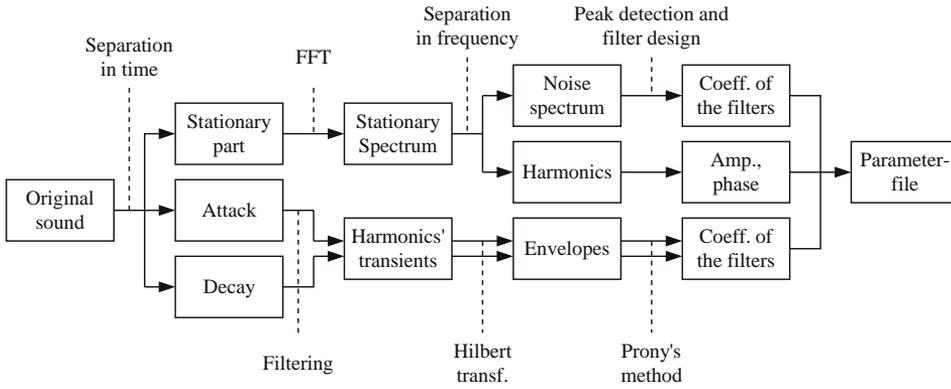
Fig. 2. The analysis process.

First, the stationary and the transient parts (the attack and the decay) were separated in the time-domain. From the stationary part the fundamental frequency and the magnitudes of the harmonic components can be readily calculated.

A novelty of the introduced method (first proposed in [15]) is that for data and computation-time reduction the attack and decay envelopes of the harmonics are implemented as step responses of IIR-filters. Using this method, the kth harmonics at time step n can be computed as

$$y_{k,n} = h_{k,n} A_k \cos(2\pi(kf_0/f_s)n + \varphi_k), \qquad (k = 1 \ldots N), \qquad (1)$$

where $y_{k,n}$ is the harmonic component, A_k and φ_k are the relative magnitude and phase of the component, f_0 and f_s are the fundamental and the sampling frequency, respectively, and $h_{k,n}$ represents the samples of the step-response of the designed envelope-filter.

The envelope-generator filter can be designed as follows. After isolating all the harmonics by FIR filtering, the envelopes can be determined as the absolute value of the harmonics' analytical signal which is a complex signal originated from the real signals and their Hilbert-transform [16]. To get the best time-domain result, the obtained envelopes were smoothed, and a step-response of a 2nd or 3rd order IIR filter was fitted on each of them using Prony's time-domain filter design method [17].

To realize the important wind-noise, a noise-filter was designed as follows. After subtracting the discrete components from the spectrum, 2nd order resonant filters were fitted to the specific peaks (given by the center frequency, gain level and damping factor) in the averaged noise spectrum. The resulted filter consists of 6-10 2nd order resonators.

The examined external effects were only the reverberation of the hall and the location of the pipes. This latter one can be modelled by intensity and time-delay stereo soundfield, while the reverberation can be simulated using hall-simulators.

2.4. Synthesis Results
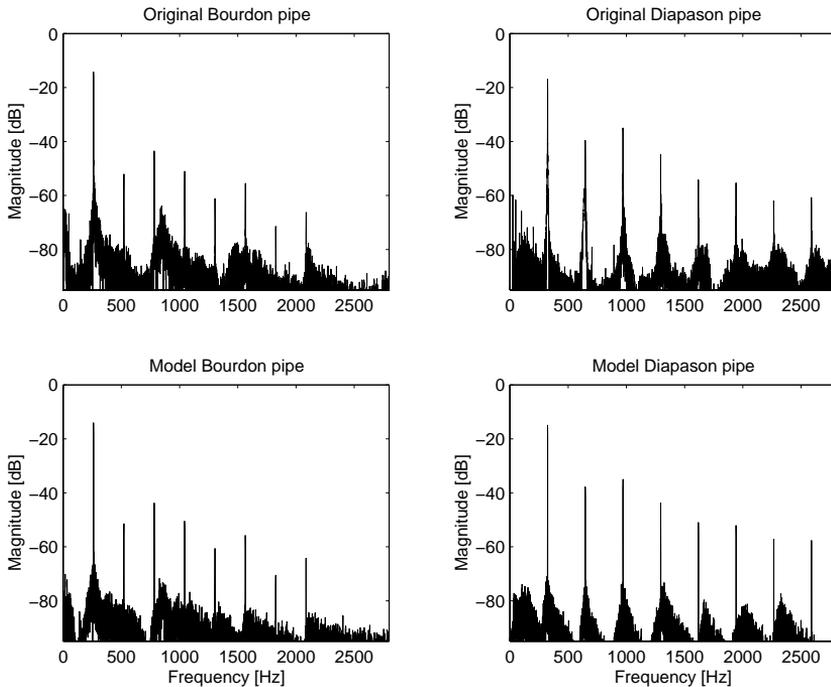
Fig. 3. The stationary spectrum of two original pipes and their models.

The spectrum of two organ pipes and their models can be seen in *Fig. 3*. The first one is a C_4-pipe of a Bourdon register (closed, wooden pipe), the second is a Diapason E_4-pipe, which is an open organ-metal pipe. It can be seen clearly that both the original and model Bourdon pipe have more noise, and their odd harmonics have smaller magnitude, than those of the Diapason pipe. Furthermore, the metal pipe has much more relevant components than the wooden one.

An example of the averaged original attack transients and the estimated 3rd order IIR filter step-responses can be seen in *Fig. 4*. The higher the order of the component, the smaller its signal-to-noise ratio, this is why the modelling is worse for higher order components. Note that their precise synthesis is not required, according to their small magnitude (cf. *Fig. 3*).

To test the efficiency of the introduced synthesis method, both an off-line MATLAB implementation and a real-time (16-bit Digital Signal Processor) implementation have been examined. For comparison, original and synthesized samples are available through the Internet, at [18].
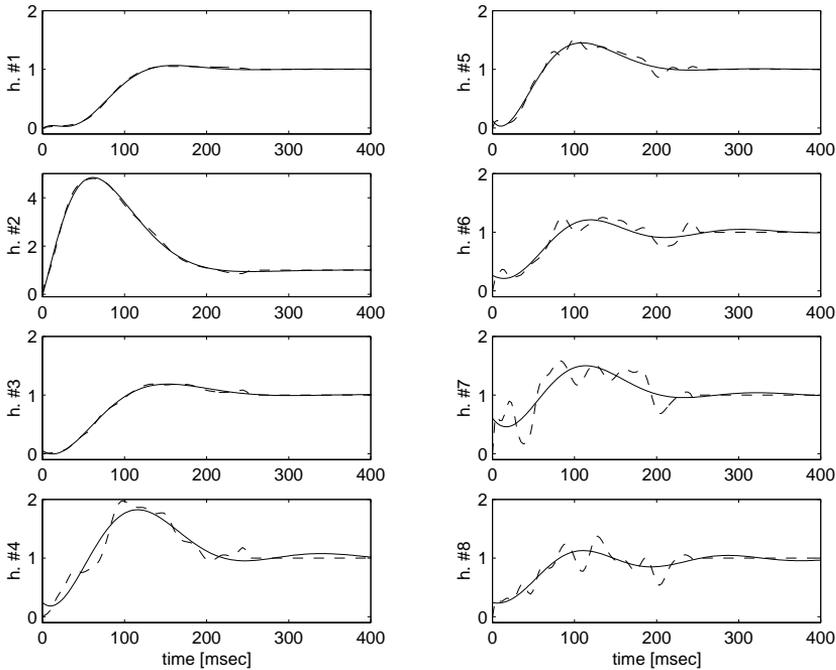
Fig. 4. The envelopes of the first 8 harmonics of a Diapason pipe (dashed lines), and the fitted step-responses (solid lines).

3. Physical Modelling

3.1. Model Structure

Since the physical modelling approach simulates the structure of the instrument, the parts of the model correspond to the parts of real instruments. In every string instrument, the heart of the sound production mechanism is the string itself. The string is excited by the excitation mechanism, which corresponds to the hammer strike in the case of the piano, or to the bow in the case of the violin. The string is responsible for the generation of the periodic sound by storing this vibrational energy in its normal modes. One part of this energy dissipates and an other part is radiated to the air by the instrument body. The body can be seen as an impedance transformer between the string and the air, which increases the effectiveness of radiation significantly. The body provides a terminating impedance to the string, therefore it also influences the modal parameters of string vibration, i.e., partial frequencies, amplitudes, and decay times. The model structure is displayed in *Fig. 5*.
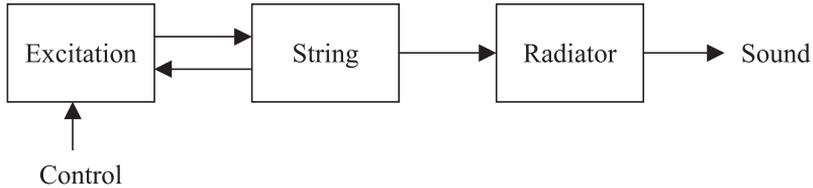
Fig. 5. The structure of the physical model.

3.2. String Modelling

A very efficient technique has been presented for string modelling in [19, 7]. The digital waveguide modelling is based on the discretisation of the solution of the wave equation. Every travelling wave which retains its shape is a solution of the wave equation. Coming from the linearity of the string, the general solution is a superposition of two travelling waves; one of them going to the right, the other to the left direction [8]:

$$y(x, t) = f^+(ct - x) + f^-(ct + x) \tag{2}$$

This equation holds for other wave variables (velocity, force, curvature) too. The digital waveguide model of the ideal string is obtained by sampling *Eq.* 2) temporally and spatially in a way that the two travelling waves move one spatial sampling interval during one time-instant [7]. This is implemented by two parallel delay lines, where the transversal displacement of the string is calculated by adding up the output of the samples of the two delay lines at the same spatial coordinate.

The termination of the string can be modelled by connecting the two delay lines at their endpoints. An ideally rigid termination corresponds to a multiplication of -1, meaning that the travelling waves are reflected with a sign change. In practice, the string is terminated by a frequency dependent impedance, introducing losses to the string vibration. This is taken into account by a digital filter $H_r(z)$ at the end of the delay line. Moreover, the distributed losses and dispersion of the string can also be approximated by the lumped reflection filter $H_r(z)$ [7]. *Fig.* 6 displays the digital waveguide model in its physical form, where M represents the length of the string in spatial sampling intervals, M_{in} denotes the position of the force input, and $H_r(z)$ refers to the reflection filter.

3.2.1. Reflection Filter Design

The impulse response of the digital waveguide is a quasi-periodic set of exponentially decaying sinusoids, whose frequencies and decay times can be controlled by the careful design of the reflection filter $H_r(z)$. In practice, the model parameters are
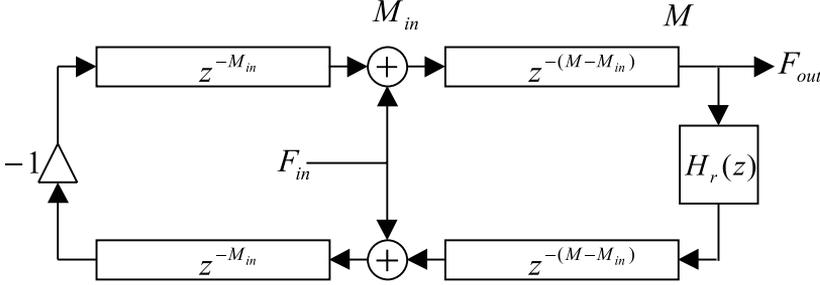
Fig. 6. The digital waveguide with consolidated losses and dispersion realized by the filter $H_r(z)$. The $z^{-M_{in}}$ and $z^{-(M-M_{in})}$ stand for delay lines of M_{in} and $M - M_{in}$ samples, respectively. The hammer force is referred by F_{in} and the force at the bridge is referred by F_{out}.

estimated from recorded tones, since that requires the measurement of one signal only. The partial frequencies produced by the digital waveguide of *Fig. 6* are determined by the phase response of the reflection filter $H_r(z)$, together with the total length of the delay lines. On the other hand, the decay times of the partials are influenced by the magnitude response of the loss filter. Therefore, it is straightforward to split the design process into three independent parts: $H_r(z) = -H_l(z)H_d(z)H_{fd}(z)$, where $H_l(z)$ is the loss filter, $H_d(z)$ is the dispersion filter, and the fractional delay filter $H_{fd}(z)$ is required for fine-tuning the fundamental frequency of the string.

The role of the loss filter $H_l(z)$ is to set the decay times of the partials. Therefore, the decay times of the recorded tone should be estimated, based on the amplitude envelopes of the partials [20]. The partial envelopes can be calculated either from analytical signals (cf. Section 2), or with heterodyne filtering [21] or with sinusoidal peak tracking utilizing Short Time Fourier Transform [20]. As the nearly exponential decay becomes approximately linear using logarithmic amplitude scale, the decay time and initial amplitude parameters can be estimated by linear regression [20, 21].

The specification for the loss filter can be computed as follows:

$$g_k = \left| H_l \left(e^{j\vartheta_k} \right) \right| = e^{-\frac{k}{f_k \tau_k}}, \tag{3}$$

where τ_k is the decay time of partial k, f_k is the frequency of partial k and g_k is the desired amplitude value of the loss filter at the angular frequency ϑ_k. Fitting a filter to g_k coefficients is not trivial, even if the phase part of the transfer function is not considered, as the error in the decay time is a nonlinear function of the amplitude error. When the magnitude response of the loss filter exceeds unity, the stability of the feedback loop is at risk.

Instead of designing the loss filter with respect to magnitude error, [19] suggests to optimize the loss filter with respect to decay times. We have also developed

filter design algorithms based on the decay-time error [22, 23]. Decay-time based optimization assures that the overall decay time of the note is preserved and the stability of the feedback loop is maintained, as negative decay times can be excluded. Moreover, optimization with respect to decay times is perceptually more meaningful.

Founded on these ideas, we have proposed a simple and robust method for high-order loss-filter design based on a special weighting function [23]. The resulting decay times of the digital waveguide are computed from the magnitude response $\hat{g}_k = |H(e^{j\vartheta_k})|$ of the loss filter by the inverse of *Eq.* (3):

$$\hat{\tau}_k = d(\hat{g}_k) = -1/(f_0 \ln \hat{g}_k) \tag{4}$$

If the function $d(\hat{g}_k)$ is approximated by the first-order Taylor polynomial around the specification g_k, the mean-square error with respect to decay times is obtained by:

$$e_\tau = \sum_{k=1}^{K} (\hat{\tau}_k - \tau_k)^2 = \sum_{k=1}^{K} (d(\hat{g}_k) - d(g_k))^2 \approx \tag{5}$$

$$\approx \sum_{k=1}^{K} (d'(g_k)(\hat{g}_k - g_k))^2 = \sum_{k=1}^{K} w_k (\hat{g}_k - g_k)^2 \tag{6}$$

which is a simple mean-squares minimization with weights $w_k = (d'(g_k))^2$, and can be done by any standard filter design technique.

The first derivative of $d(g_k)$ is $d'(g_k) = 1/(f_0 g_k (\ln g_k)^2)$, which can be approximated by $d'(g_k) \approx 1/(f_0(g_k - 1)^2)$. Since $1/f_0$ does not depend on k, it can be omitted from the weighting function. Hence, the weighting function becomes:

$$w_k = \frac{1}{g_k^2 (\ln g_k)^4} \approx \frac{1}{(g_k - 1)^4}. \tag{7}$$

The phase specification of the loss filter is computed by the Hilbert transform [16] from the interpolated magnitude specification, corresponding to a minimum-phase filter.

Fig. 7 depicts the results of loss filter design for a filter order of 8 with solid line. The measured decay times of the piano note $A_4^\sharp$ are displayed with crosses. The resulted decay times using a one-pole loss filter designed by polynomial regression [22] are displayed with dashed line. Although the general trend and the decay times of the first ten partials are already modelled precisely by the one-pole loss filter, in some cases it is still advantageous to use higher-order loss filter. We have used 3^{d} order loss filters for piano modelling, and one-pole loss filters for the modelling of the violin. This distinction is motivated by the fact that the piano has a decaying tone, therefore the decay rates have a great perceptual importance, while the violin is a continuously excited instrument, where the precise rendering of the decay rates are less significant for the listeners.
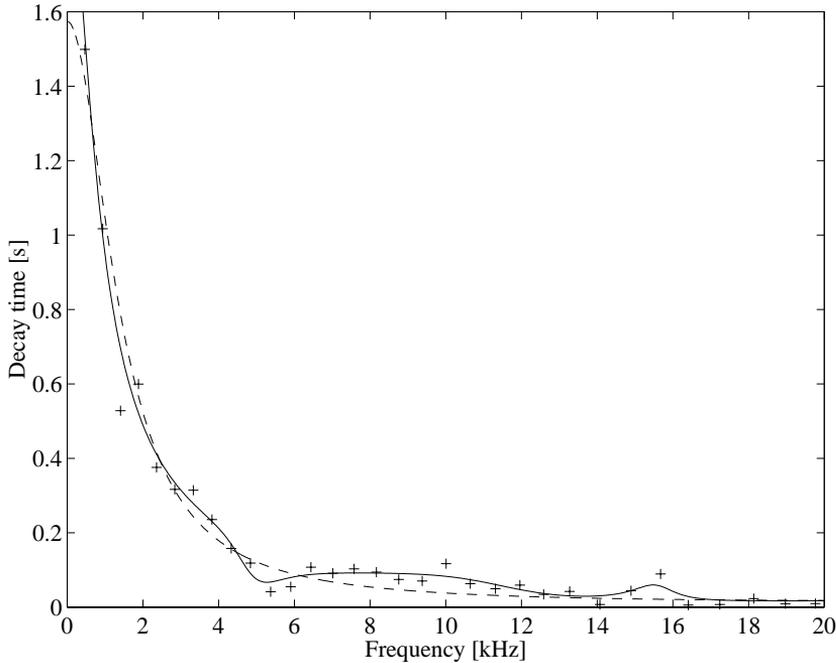
Fig. 7. Loss filter design for $A_4^\sharp$ piano note: prescribed decay times (crosses), the decay times approximated by the one-pole loss filter of [22] (dashed-line), and by an 8th order loss filter designed by the method based on a special weighting function (solid line).

In the case of piano modelling, the audible effect of string dispersion cannot be neglected. Dispersion denotes an increase in wave velocity for higher frequencies. This can be modelled by having a 'shorter' delay line for the higher partials than for the lower ones. For that, a filter with a decreasing phase delay is required. Since the magnitude response of the reflection filter $H_r(z) = -H_l(z)H_d(z)H_{fd}(z)$ should only be affected by the loss filter $H_l(z)$, it is straightforward to use an allpass filter as dispersion filter $H_d(z)$. For the design, we have used the method based on an iterative least-squares algorithm [24, 25]. A filter order $N = 16$ has been required for the lowest piano notes to provide good results, while for the middle register a fourth-order dispersion filter has been found to be enough. For the violin, the dispersion is negligible, therefore the dispersion filter $H_d(z)$ does not need to be implemented.

The string needs to be fine tuned because delay lines can implement only an integer delay and this provides too low a resolution for the fundamental frequencies. Fine tuning can be incorporated in the dispersion filter design or, alternatively, a
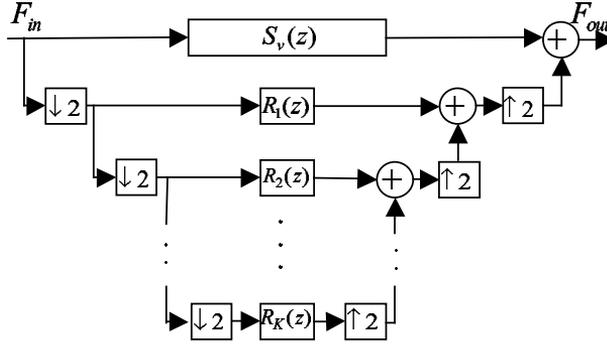
Fig. 8. The multi-rate resonator bank. The basic digital waveguide is referred by $S_v(z)$ and
$R_1(z) \ldots R_K(z)$ stands for the parallel resonators. Downsampling and upsampling
operations are marked by $\downarrow 2$ and $\uparrow 2$, respectively.

separate fractional delay filter $H_{fd}(z)$ can be used in series with the delay line. In
this study, we have used a first-order allpass filter for this purpose. Another type of
fractional delay filters could be also used, [26] provides an exhaustive overview on
their design.

3.2.2. Modelling Beating and Two-Stage Decay

In real pianos, except for the lowest octave, the vibration of two or three strings
are coupled through the bridge, when one note is played. This produces beating
and two-stage decay in the sound [27]. This effect can be simulated by having two
coupled waveguides in parallel [28], but this leads to high computational cost and
complicated parameter estimation.

Instead, we suggest to use some second-order resonators $R_1 \ldots R_K$ parallel
with the string model $S_v(z)$ [22, 29]. This is depicted in *Fig. 8*. The transfer function
of the resonators $R_k(z)$ are as follows:

$$R_k(z) = \frac{\text{Re}\{a_k\} - \text{Re}\{a_k \overline{p_k}\}z^{-1}}{1 - 2\,\text{Re}\{p_k\}z^{-1} + |p_k|^2 z^{-2}}$$

$$a_k = A_k e^{j\varphi_k} \qquad p_k = e^{j\frac{2\pi f_k}{f_s} - \frac{1}{f_s \tau_k}}, \tag{8}$$

where A_k, φ_k, f_k, and τ_k refer to the initial amplitude, initial phase, frequency and
decay time parameters of the kth resonator, respectively. The overline stands for
complex conjugation, the *Re* sign for taking the real part of a complex variable, and
f_s is the sampling frequency.

The resonators are tuned close to the frequency of the distinct partials of the digital waveguide. Thus, every partial corresponds to two slightly mistuned sinusoids with different decay times and amplitudes, and their superposition produces beating and two-stage decay. The efficiency of the structure comes from the fact that only those partials have parallel resonators, where the beating and two-stage decay are prominent. The others have simple exponential decay determined by the digital waveguide model $S_v(z)$.

If the resonators are implemented in a multi-rate manner [29], the method provides significant computational savings compared to having a second waveguide in parallel (5–10 operations/cycle instead of 30–40). Moreover, the parameter estimation simplifies to finding the parameters of the mode-pairs. The stability problems of a coupled system are also avoided.

3.2.3. Finger Modelling

On the violin, the player has to use his fingers to change the length of the strings and thus to change the fundamental frequency of the tone. These note transitions are important in determining the characteristics of the instrument. Physically the finger acts like a damper attached to the string, which can be modelled by a scattering junction with variable position. The frequency-dependent low pass filtering effect of the finger can be realized within the reflection filter $H_r(z)$ as well. The scattering junction is similar to finger hole models in woodwinds [30].

In our experiments we have used a simplified junction combined with a simple fractional delay for fine tuning the frequency of the sound (see *Fig. 9*). With the increase of the pressure of the finger (p), the right side of the delay lines gets less signal. Finally, when $p = 1$, the shortened left side string is terminated properly with -1 (and tuned with the fractional delay D).
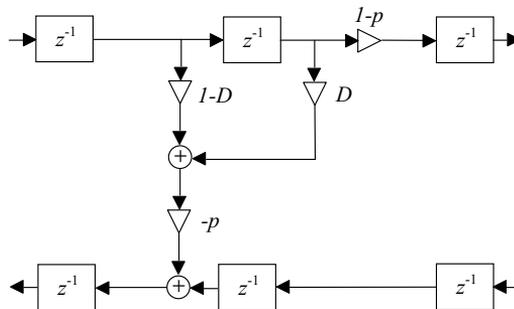
Fig. 9. Simplified model of a string terminated by a finger. The z^{-1} sign stands for a unit delay, $0 \leq p \leq 1$ is the finger pressure coefficient, and $0 \leq D \leq 1$ stands for the length of the fractional delay.

The finger model described above models only the transitions from an open string note to a finger-terminated one and vice versa. However, in most cases the player uses another finger to change from one note to the other, therefore two finger junctions need to be included in the model. In practice, two types of transitions have to be simulated depending on the direction of change. Changing to a higher note requires that the first finger is already on the string and the second one is being used normally, with increasing finger pressure. Changing to a lower note may assume that the second finger is already in its place (behind the other) while the pressure of the first finger is lowered to zero. By properly choosing the shape and the speed of the pressure change several effects can be modelled.

Furthermore, differences in the four strings of the violin can be considered to refine the model. Each string has its own properties (fundamental frequency, specific impedance, stiffness, etc.), thus, each has different tone. The player has the flexibility of choosing a string for a given note. The decision depends on the pitch of the actual note, the notes following and preceding the actual one and the timbre the musician wants to achieve. When a new note is started on a new string, a previously excited open string or finger-terminated string might still vibrate, or the latter might change to open string (if the player lifts away his finger). When off-line synthesis is used, these subtleties can be set individually for each tone manually, or transition rules can be formed to take them into account. On the contrary, for real-time synthesis only general rules can be used for easier controllability.

3.3. Body Modelling

The radiation of the soundboard or any instrument body is generally treated as a linear filtering operation acting on the string signal. Thus, body modelling reduces to filter design. Theoretically, this filter should be somewhat different for all the strings. This is feasible for the four strings of the violin, but for modelling the piano having hundreds of strings, this would lead to very high computational load, which is not acceptable. In practice, the string signals are summed and lead through a single body filter to reduce the required computational complexity.

Instrument bodies exhibit a high modal density, therefore high-order filters are needed for their simulation. In the case of the guitar body, the required filter order was about 500 [31]. We have found that the piano requires even higher filter orders. In the case of FIR filters, 2000 taps were necessary to provide high quality sound. Commuted synthesis [32] could overstep this problem, but that would require simplifications in the excitation model. Feedback delay networks [33] are capable of producing high modal density at a low computational cost, but due to the difficulties in parameter estimation, they have not been used for high-quality sound synthesis.

To resolve this problem, we have proposed a novel multi-rate approach for instrument body modelling [34]. The body model is depicted in *Fig. 10*. The string signal F_s is split into two parts: the lower is downsampled by a factor of 8 and

filtered by a high order FIR filter $H_{\text{low}}(z)$, precisely synthesizing the instrument body response up to 2.2 kHz. Above 2.2 kHz, only the overall magnitude response of the body is modelled by a low order FIR filter $H_{\text{high}}(z)$. This signal is delayed by N samples to compensate the delay of the decimation and interpolation filters of the low frequency chain. We have found that a downsampling factor of 8 is a good compromise between sound quality and computational efficiency.
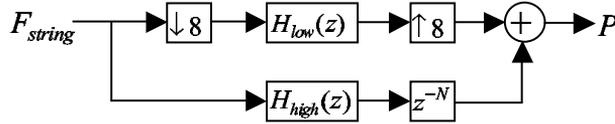
Fig. 10. The multi-rate body model. The body modes below 2.2 kHz are precisely synthesized by the filter $H_{low}(z)$ running at a lower sampling rate, while the higher modes are approximated by a low order filter $H_{high}(z)$. Down- and upsampling by a factor of 8 are referred by $\downarrow 8$ and $\uparrow 8$, respectively.

As an example, the magnitude response of a piano soundboard model is depicted in *Fig. 11*. It can be seen in the figure that the magnitude response is accurately preserved up to 2 kHz. Although not displayed, but so is the phase response. Above 2 kHz, only the overall magnitude response is retained. The proposed model is capable to produce high sound quality at around 100 instructions per cycle and provide a very similar sonic character compared to the original 2000 tap FIR filter.

3.4. Excitation Modelling

The string and body models are of the same structure for the different string instruments, only their parameters are different. On the contrary, the excitation models differ for the violin and the piano, as their excitation mechanisms are completely different, and their precise implementation is essential for rendering the sonic characteristics of these instruments.

3.4.1. The Hammer Model

The piano string is excited by a hammer, whose initial velocity is controlled by the player with the strength of the touch on the keys. The excitation mechanism of the piano is as follows: as the hammer hits the string, the hammer felt compresses and feeds energy to the string, then the interaction force pushes the hammer away from the string. Accordingly, the excitation is not continuous, it is present for some milliseconds only. The hardwood core of the hammer is covered by wool felt,
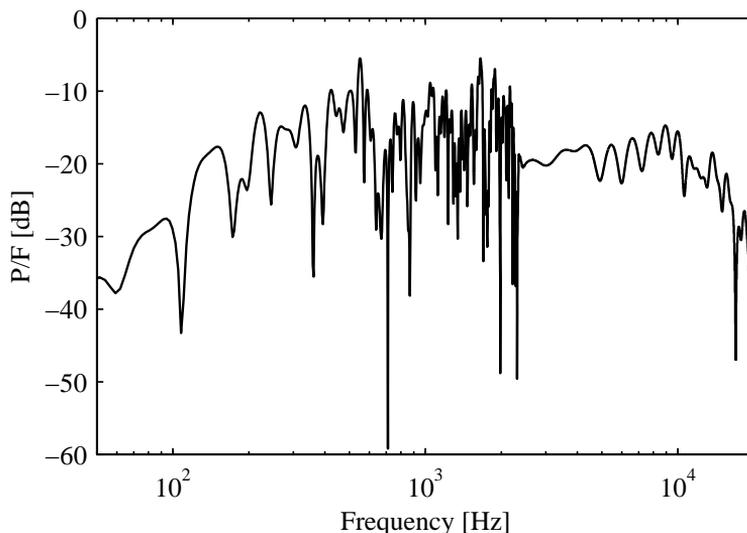
Fig. 11. The magnitude transfer function of a multi-rate soundboard model.

whose hardness increases in the function of compression. This is the reason why playing harder on the piano results not only in a louder tone, but also in a spectrum with stronger high-frequency content.

The piano hammer is generally modelled by a small mass connected to a nonlinear spring [35]. The equations describing the interaction are as follows:

$$F(t) = f(\Delta y) = \begin{cases} K(\Delta y)^p & \text{if } \Delta y > 0 \\ 0 & \text{if } \Delta y \leq 0 \end{cases} \qquad (9)$$

$$F(t) = -m_h \frac{d^2 y_h(t)}{dt^2}, \qquad (10)$$

where $F(t)$ is the interaction force, $\Delta y = y_h(t) - y_s(t)$ is the compression of the hammer felt, where $y_h(t)$ and $y_s(t)$ are the positions of the hammer and the string, respectively. The hammer mass is referred by m_h, K is the hammer stiffness coefficient, and p is the stiffness exponent.

These equations can be easily discretized with respect to time. However, the straightforward approach (assuming $F(t_n) \approx F(t_{n-1})$) can lead to numerical instabilities for high impact velocities. This can be avoided by rearranging the nonlinear equations to known and unknown terms [36].

We have suggested a simpler approach for avoiding the numerical instability [37]. The proposed multi-rate hammer model is depicted in *Fig. 12*. If the continuous-time system is stable, the stability of the discrete system can always be assured with a sufficiently high sampling rate f_s, as for $f_s = \infty$, it will behave
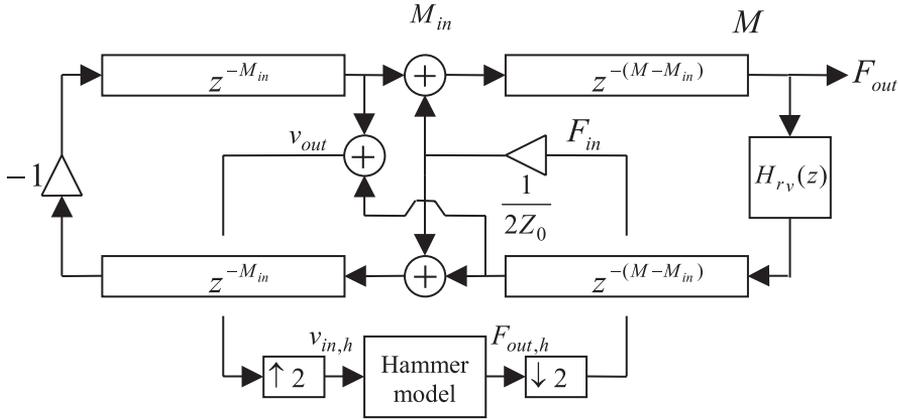
Fig. 12. The digital waveguide model of *Fig. 6* connected to the hammer running at a double sampling rate ($\downarrow$ 2 and $\uparrow$ 2 stand for down- and upsampling). The string impedance is referred by Z_0.

as the continuous-time equations. However, increasing the sampling rate of the whole model would lead to unacceptable computational overhead. When only the sampling rate of the hammer model is increased, it leads to a small computational overhead, while still assures that $F(t_n)$ changes a little at every time-step. Implementing the hammer at a double sampling rate has been found to provide stable results. For downsampling ($\downarrow$ 2 in *Fig. 12*) simple averaging, for upsampling ($\uparrow$ 2 in *Fig. 12*) linear interpolation is used.

3.4.2. The Bow Model

In the case of bowed instruments the excitation is based on the sticking friction between the string and the bow hairs. The bow, moving perpendicularly to the string, grips the string (gripping phase). This friction force is highly nonlinear. Due to the increasing displacement of the string, the elastic returning force is also increasing until its level reaches the sticking friction. At this point the bow releases the string, the string swings back (release phase) and then vibrates freely. This vibration is damped partly by the own losses of the string and partly by the slipping friction that develops between the string and the bow hairs. This state lasts as long as the bow grips the string again, which occurs only when the velocity of the bow and the string equals. In this case, their relative velocity is zero, the frictional force is maximal. This alteration of the stick and slip phases is the so-called Helmholtz motion. The excitation is periodical and generates a sawtooth-shaped vibration.

The excitation depends on several control variables. The primary control variable is the velocity of the bow, other important factors are the force of the bow exerted on the string and the position of the bow along the string. Less important variables are the angle between the bow and the string, the size of the contact surface of the bow, and the grip of the bow hair (which can be increased by rosin). In order to keep the model manageable and realizable, usually only the primary and some other important variables (such as the bow force and position) are taken into account.

The bow-string interaction is usually modelled by a scattering junction [38] (*Fig. 13*). This junction is controlled by differential velocity (v_Δ^+), which is the difference of the bow velocity and the current string velocity. The position of bowing determines the insertion point of the junction into the delay lines. Other control variables (bowing force and angle, etc.) are changed by modifying the parameters of the reflection function ($\rho(v_\Delta^+)$). This function also depends on the characteristic impedance of the string and on the friction coefficient between the bow and the string.
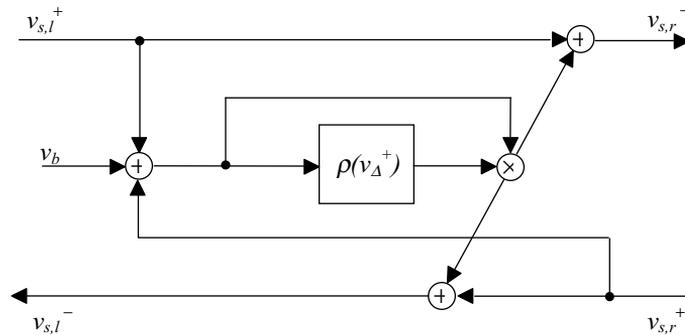
Fig. 13. The scattering junction for modeling the bow-string interaction. The incoming and outgoing wave velocities for the left-hand part of the string are referred by $v_{s,l}^+$ and $v_{s,l}^-$. Similar notation is used ($v_{s,r}^+$ and $v_{s,r}^-$) for the right-hand part, too. The reflection function is referred by $\rho(v_\Delta^+)$ and v_b stands for the bow velocity.

Besides modelling the bow-string interaction, the player has to be modelled as well. The problem of modelling the left hand was discussed in Section 3.2.3. An exact model of the right (bowing) hand should provide enormous degrees of freedom using interactive controllers. This would result again an unmanageable instrument, and/or it would require a real violin player at the control keyboard/violin. Similarly to the proposed finger model, this problem can also be resolved by an automatic system based on real playing styles on bowed instruments. For each bowing style the time variations of the primary control variables can be represented by characteristic envelopes, so only one parameter needs to be adjusted for a given style. A MIDI based implementation of this idea can be found in [39].

4. The Comparison of the Two Synthesis Methods

Here we compare the two methods described in this paper, namely the signal modelling based on envelope-filters and the physical modelling based on digital waveguides. When mentioning signal modelling and physical modelling throughout this section, we are referring to these two models covered in the paper. As our signal model describes the partial envelopes by linear filters, even theoretical connections can be found between the two methods. The theoretical investigations are followed by practical considerations.

4.1. Theoretical Connections

We show that the impulse response of both formulations can be expressed as a sum of exponentially decaying sinusoids, which can be realized as a resonator bank. Naturally, the resonator bank implementation is not an efficient realization, its only purpose is to serve as a common base for the comparison of the two methods. We show that for certain circumstances the two modelling approaches produce the same output signal.

4.1.1. The Signal Model

Recalling *Eq.* (1), the signal model was based on the idea of switching on a sine wave when a note is played and multiplying it with the attack and decay envelope of the given harmonics:

$$y_{k,n} = h_{k,n} A_k \cos(2\pi (k f_0 / f_s) n + \varphi_k), \ (k = 1..N) \tag{11}$$

Here the attack envelope $h_{k,n}$ is realized as step responses of 2nd or 3rd order filters.
The step response can be further rewritten as

$$h_{k,n} = w_{k,n} * \varepsilon_n, \tag{12}$$

where $w_{k,n}$ is the impulse response of the filter, ε_n is the step function and $*$ denotes convolution. The main effects of *Eq.* (11) in the time domain are depicted in *Fig. 14*.
Multiplication in the time domain with a sine wave is a simple modulation. Hence, in the frequency domain it becomes the convolution of the sine wave and the step response of the envelope filter, i.e.

$$Y(z) = (W(z)E(z)) * X(z). \tag{13}$$

In this special case, this equation can be rewritten as follows:

$$Y(z) = (W(z) * X(z))(E(z) * X(z)) = R(z)(E(z) * X(z)). \tag{14}$$
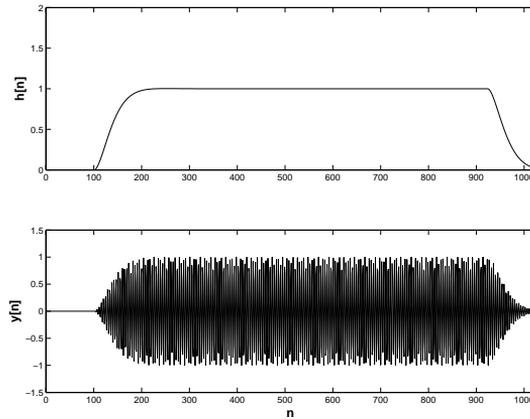
Fig. 14. Signal model of a given partial realized with envelope filters. (a) step response of the envelope filter($h[n]$); (b) output of the system ($y[n] = h[n]x[n]$).

Note that $R(z) = W(z) * X(z)$ in the time domain is $r[n] = w[n]x[n]$, i.e. a sine wave multiplied with a second order system's impulse response. In the frequency domain, the convolution with the sine wave shifts up the original filter poles located at DC to the frequency of the sine wave. Thus, this expression can be realized with the same number of resonators as the number of poles of the original filter. The input to these resonators is the sine wave triggered by the trigger signal $\varepsilon[n]$. *Fig.15* shows the time-domain signals of this realization.
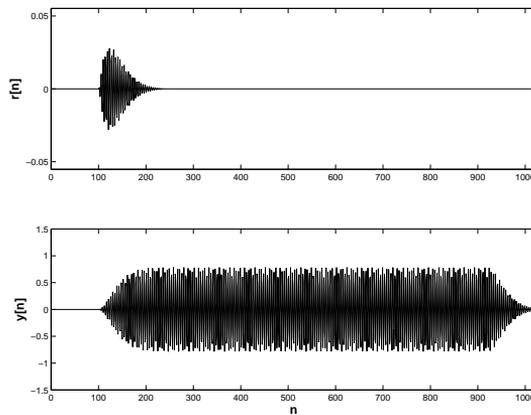
Fig. 15. Signal model of a given partial realized with resonators. (a) impulse response of the two resonators $r[n]$; (b) output of the system ($r[n]$ excited by a sinusoid).

Thus, the signal model with envelope filters applied to the partials of the sound

can be realized with a set of resonators. The required number of resonators depends on the number of partials to be generated and on the order of the envelope filters.

4.1.2. The Physical Model

The transfer function of the digital waveguide model of *Fig.* 6, assuming that the reflection filter is constant for all the frequencies (i.e, $H_r(z) = -r, 0 < r < 1$) is:

$$\frac{F_{\text{out}}}{F_{\text{in}}} = \frac{1}{1 - rz^{-N}} \left(1 + z^{-2M_{\text{in}}}\right) z^{-(M - M_{\text{in}})} \tag{15}$$

After the fractional expansion of the denominator of *Eq.* (15) we obtain the transfer function of a set of complex exponentials:

$$\frac{F_{\text{out}}}{F_{\text{in}}} = \left\{ \frac{a_1}{1 - z^{-1}r_1 e^{j\vartheta_1}} + \ldots + \frac{a_N}{1 - z^{-1}r_N e^{j\vartheta_N}} \right\}$$

$$a_k = j\frac{2}{N} \sin(2k\pi \frac{M_{\text{in}}}{N}) e^{-j\vartheta_k M}$$

$$r_1 = \ldots = r_N = r^{\frac{1}{N}}, \tag{16}$$

where $\vartheta_k = (2k\pi)/N$ is the frequency of the kth mode, $N = 2M$ is the total length of the delay line, a_k are the complex amplitudes and r_k are the pole radii. The impulse response $h(n)$ of the digital waveguide can be obtained from *Eq.* (16) by the inverse $\mathbb{Z}$ transform:

$$h(n) = \sum_{k=1}^{N} a_k \left(r_k e^{j\vartheta_k}\right)^n = \sum_{k=1}^{N/2} a_k \left(r_k e^{j\vartheta_k}\right)^n + a_{N-k} \left(r_{N-k} e^{j\vartheta_{N-k}}\right)^n. \tag{17}$$

As $\vartheta_{N-k} = 2\pi - \vartheta_k$, the corresponding pole pairs will be conjugate pairs $r^{N-k} e^{j\vartheta_{N-k}} = r^k e^{-j\vartheta_k}$, and so the amplitudes $a_{N-k} = \overline{a_k}$, where the overline refers to complex conjugation. Therefore the impulse response $h(n)$ can be expressed as a sum of exponentially decaying sinusoids:

$$h(n) = \sum_{k=1}^{N/2} r_k^N \left(a_k e^{j\vartheta_k n} + \overline{a_k} e^{-j\vartheta_k n}\right) = \sum_{k=1}^{N/2} |a_k| r_k^N \sin\left(\vartheta_k n + \varphi_k\right) \tag{18}$$

where $|a_k|$ is the magnitude, and φ_k is the phase of the complex coefficient a_k.

It can be seen from *Eq.* (18) that the impulse response of the digital waveguide with $H_r(z) = -r$ is the sum of exponentially decaying sinusoids, whose frequencies are equally distributed on the unit circle, and their decay rates are equal. For an arbitrary reflection filter $H_r(z)$ the modal frequencies and decay times cannot be derived in a closed form, however, they can be determined by numerical iterations.

In any case, the digital waveguide can always be substituted by a set of parallel resonators. Their impulse responses are exponentially decaying sinusoids with arbitrary initial amplitudes and phases, thus, they can be implemented as second order IIR filters in parallel.

Similar derivations with a different formulation were presented in [28], and it was shown that if two or three waveguides are coupled, the partials can be expressed by the sum of two or three sinusoids. Obviously, when the beating and two-stage decay of the piano is modelled by the multi-rate resonator bank of Section 3.2.2, the equivalent resonator structure can be obtained by adding the parallel resonators $R_1 \ldots R_k$ of *Fig. 8* to the equivalent resonators of the waveguide. In this case, two resonators will correspond to some of the partials.

4.1.3. The Link

So far, the digital waveguide model has been substituted by a set of resonators connected in parallel, behaving in the same way as the original string model. Now, the question is in which cases the signal model of Section 2 can produce an equivalent output compared to the digital waveguide.

In the case of the piano, the hammer excitation is impulse-like, thus, its main role is to set the initial amplitudes of the partials. After the hammer has left the string, the partial envelopes decay exponentially in the string signal (here we neglect the transients introduced by the soundboard). Therefore, for a specific hammer velocity, each partial can be modelled by a sine generator connected to an envelope-filter, as described in Section 2. Thus, in the case of the piano, the signal model produces the same output as the physical model, except the initial transients.

For the violin and for the organ the link between the physics of the instruments and the envelope-filter based signal model is not as clear as for the piano. As these two instruments are continuously excited, and their excitations are of nonlinear nature, the partials cannot be synthesized by a set of exponentially decaying sinusoids. Accordingly, the partial envelopes cannot be precisely described by linear filters. From a physical point of view, the organ pipe and also the violin can be modelled by a single digital waveguide connected to a nonlinear exciter. In our signal model approach this nonlinear system is modelled with a linear system of a higher order. Third order envelope-filters have been found to be adequate for modelling the organ sound, this is equivalent to three digital waveguides coupled to each other. In other words, three linearly excited and coupled acoustic tubes produce similar sound to one tube connected to a nonlinear exciter.

4.2. Practical Considerations

In this section, our signal-based and physics-based approach is compared, from the point of view of their applicability to different instruments.

4.2.1. The Methods

The signal-based approach models the sound of the instrument itself. Accordingly, it does not make any assumptions on the structure of the musical instrument, only that the generated sound is periodic. Therefore, it can model a wide range of instrument sounds. As it is a general representation, its parameter estimation is simple, basically reduces to tracking partial envelopes, which can be easily auto-mated. As the structure of the instrument is not modelled, the interaction of the musician cannot be easily taken into account, meaning that, e.g., for different bow forces or velocities in the case of the violin different parameter sets are required for resynthesis. In practice, this means that for a single note the analysis procedure has to be run for all the different playing styles that a player can produce, and a large amount of data has to be stored or transmitted. As it treats the notes separately, the interaction of the different notes, e.g., the coupled vibration of strings, cannot be easily modelled. The quality and the computational load of the synthesis is usually varied by changing the number of simulated partials, which is probably not the best way from a perceptual point of view.

The physics-based approach models the functioning of the instrument, rather than the produced sound itself. It makes assumptions about the instrument it models, therefore, it looses generality. Consequently, the parameter estimation cannot be completely automated, at least the model structure has to be determined by the user. As the model structure already describes the main features of the instrument, only small numbers of parameters are needed, and modifications to these parameters produce perceptually meaningful results. For example, the user now controls the bow force, rather than the loudness of a single partial, and the instrument reacts in a way as a real violin would do. Therefore, only one parameter set is required, since the different playing styles according to the interaction of the musician are automatically modelled. As it describes the physical structure, the interaction of the different model parts are also taken into account, e.g., the string coupling on the piano is easily modelled. A drawback that none of the tones will be perfectly modelled: the model may sound as a piano, but will be always different from that piano where its parameters come from. The quality and the computational load is varied by, e.g., changing the accuracy of modelling losses and dispersion, rather than changing the number of simulated partials, which is less noticeable for the listener. These characteristics are summarized in *Table 1*.

4.2.2. The Instruments

The choice between the two approaches strongly depends on which instrument should be modelled. The features which are relevant from this viewpoint for the instruments covered in this paper are listed in *Table 2*. Naturally, other factors also influence the choice of the user, e.g., if automatic parameter estimation is required, the signal modelling approach should be chosen.

Table 1. Main features of the synthesis methods described in the paper.

Method	Signal modelling	Physical modeling
Assumptions on the structure	Poor	Yes
Generality	Yes	No
Parameter estimation	Simple	Complicated
Nature of parameters	Abstract	Meaningful
Modelling a specific sound	Precise	Approximate
Interaction of the musician	Hard to modell	Modelled
Interaction of instrument parts	Hard to modell	Modelled

The sound of a specific organ pipe cannot be influenced by the player. More-over, the coupling between the different pipes is negligible, therefore the different tones can be synthesized independently. As signal modelling describes a specific sound almost perfectly, it is the best choice for organ synthesis. Its computational load is acceptable, since the number of partials is low in the case of the organ flue pipes.

In the case of the piano, the player can vary only one parameter for a given note, by changing the impact velocity of the hammer, thus, the timbre space of one note is one-dimensional. For a signal model, this would mean storing different parameter sets for a few hammer velocities, and interpolation could be used between sets. Although it is also possible with the signal model, the effect of the player is much easier modelled by the physics-based approach. Moreover, the strings of the piano are coupled when the damper pedal is depressed which is also controlled by the player: this can be modelled by the physics-based approach only. As the low piano tones may contain about hundred partials, the signal based model would be computationally more demanding than the physical model based on digital wave-guides.

For the violin, the freedom of the player is enormous: he can vary the bow force, velocity, position, and angle, the finger position and pressure, and decide on which string he plays the given note. Therefore, the timbre space of the violin is multi-dimensional: for signal-based synthesis many sounds along all these dimen-sions should be recorded and analyzed. Since the goal is not only to render the sound of a specific violin note, but to create a playable instrument, the only choice which remains is physical modelling. The inputs of the physical model are the real physical parameters (e.g., bow force and velocity), therefore the effect of the player is automatically taken into account.

Table 2. Main features of the different instruments, serving a base for choosing the proper synthesis approach.

Instrument	Organ	Piano	Violin
Number of partials	< 20	5–100	10–50
Number of playing parameters	0	Few	Many
Coupling between the instrument parts	Negligible	Present	Significant

5. Conclusion

In this paper signal-model and physical-model based sound synthesis methods have been described, namely additive synthesis with envelope-filters and digital waveguide modelling. Three case studies (applications of the methods to the sound synthesis of the organ, piano and the violin) have been introduced, and detailed analysis of the effectiveness of the different synthesis methods have been discussed.

The proposed additive synthesis method utilizing efficient and physics-base envelope generation is capable of the accurate reproduction of a specific sound of an instrument, but primarily the sound from which its parameters are derived from. The model can be made more realistic by analyzing more than one waveforms and including both the systematic and the random variations of each parameter. Realizing these instrument-specific variations, the signal model is able to behave as a real instrument. However, as the parameters of the model are not related directly with those of the instrument, the efficient control of the model is not an easy task.

As the physical model is based on the physics of real instruments, its transient and random behavior is close to those of the instrument to be modelled. In addition, its parameters are derived directly from those of the instrument (such as string length, bow velocity), thus, controlling a physics-based instrument is a much easier task. In this paper, computationally efficient physical modelling based methods were presented. It was shown that the models need to be evaluated also from a perceptual point of view and this way the trade-off between efficiency and high fidelity can be controlled.

As a theoretical result, it was shown that the signal model and the physical model can be equivalent under specific circumstances. Finally, we have seen that all methods can be used for realistic instrument modelling, but their computational efficiency varies according to the function of the instrument to be modelled.

Acknowledgement

References

[1] SMITH, J. O., Viewpoints on the History of Digital Synthesis, *Proc. Int. Computer Music Conf.*, Montreal, Canada, September 1991, pp. 1–10.

[2] CHOWNING, J. M., The Synthesis of Complex Audio Spectra by Means of Frequency Modulation, *J. Aud. Eng. Soc.*, **21** No. 7 (1973), pp. 526–534.

[3] LE BRUN, M., Digital Waveshaping Synthesis, *J. Aud. En. Soc.*, **27** No. 4 (1979), pp. 250–266.

[4] ARFIB, D., Digital Synthesis of Complex Spectra by means of Multiplication of Nonlinear Distorted Sine Waves, *J. Aud. Eng. Soc.*, **27** No. 10 (1979), pp. 757–768.

[5] ROADS, C., *The Computer Music Tutorial*, The MIT Press, Cambridge, Massachusetts, USA, 1995.

[6] SERRA, X. – SMITH, J. O., Spectral Modelling Synthesis: a Sound Analysis/Synthesis System Based on Deterministic plus Stochastic Decomposition, *Computer Music J.*, **14** No. 4 (1990), pp. 12–24.

[7] SMITH, J. O., Physical Modeling using Digital Waveguides, *Computer Music J.*, **16** No. 4 (1992), pp. 74–91.

[8] FLETCHER, N. H. – ROSSING, T. D., *The Physics of Musical Instruments*, Springer-Verlag, New York, USA, 1998.

[9] ANGSTER, J., Modern Methods and their Results in Measurement of the Sound and Vibration of the Organ Llue Pipes, Ph.D. thesis, MTA MMSZ Acoustic Research Lab, Budapest, 1990, (In Hungarian).

[10] PISZCZALSKI, M. – GALLER, B. – BOSSEMEYER, R. – HATAMIAN, M. – LOOFT, F., Performed Music: Analysis, Synthesis, and Display by Computer, *Journal of the Audio Engineering Society of America*, **29** No. 1–2 (1981), pp. 38–46.

[11] MEYER, J., Temporal Fine Structure of Organ Sounds in Churches, In: *Proc. of the 138th Meeting of the Acoustical Society of America*, Columbus, Ohio, USA, 1–5 November 1999, p. 2200, Proceedings is Available As the *Journal of the Acoustical Society of America*, **106** No. 4, Pt. 2.

[12] ANGSTER, J. – ANGSTER, J. – MIKLÓS, A., Coupling Between Simultaneously Sounded Organ Pipes, In: *94th AES Convention (Preprint 3534)*, Berlin, Germany, 16–19 March 1993, pp. 1–8.

[13] KLOTZ, H., *Das Buch von der Orgel*, Bärenreiter Verlag, Kassel, Germany, 1960, 2000, English translation: *The Organ Handbook*, by Gerhard Krapf, Concordia Publication House, USA, 1969.

[14] PÉCELI, G., A Common Structure for Recursive Discrete Transforms, *IEEE Transactions on Circuits and Systems*, **33** No. 10 (1986), pp. 1035–1036.

[15] MÁRKUS, J., Signal Model Based Synthesis of the Sound of Organ Pipes, M.S. thesis, Budapest University of Technology and Economics, June 1999, 120 p., in Hungarian.

[16] OPPENHEIM, A. V. – SCHAFER, R. W., *Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, New Jersey, USA, 1975.

[17] BURRUS, C. S. – PARKS, T. W., Eds., *Digital Filter Design*, John Wiley & Sons, Inc., New York, October 1987.

[18] MÁRKUS, J., Signal Model Based Synthesis of the Sound of Organ Pipes, Tech. Rep., Budapest University of Technology and Economics, Budapest, Hungary, 2000, http://www.mit.bme.hu/^markus/projects/organ.

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

[20] VÄLIMÄKI, V. – HUOPANIEMI, J. – KARJALAINEN, M. – JÁNOSY, Z., Physical Modelling of Plucked String Instruments with Application to Real-Time Sound Synthesis, *J. Aud. Eng. Soc.*, **44** No. 5 (1996), pp. 331–353.

[21] VÄLIMÄKI, V. – TOLONEN, T., Development and Calibration of a Guitar Synthesizer, *J. Aud. Eng. Soc.*, **46** No. 9 (1998), pp. 766–778.

[22] BANK, B., Physics-Based Sound Synthesis of the Piano, M.S. thesis, Budapest University of Technology and Economics, Hungary, May, 2000, Published as Report 54 of HUT Laboratory of Acoustics and Audio Signal Processing, http://www.mit.bme.hu/^bank.

[23] BANK, B. – VÄLIMÄKI, V., Robust Loss Filter Design for Digital Waveguide Synthesis of String Tones, *IEEE Signal Processing Letters*, **10** No.1 (2003), pp. 18–20.

[24] ROCCHESSO, D. – SCALCON, F., Accurate Dispersion Simulation for Piano Strings, In: *Proc. Nordic. Acoust. Meeting*, Helsinki, Finland, 1996, pp. 407–414.

[25] LANG, M. – LAAKSO, T. I., Simple and Robust Method for the Design of Allpass Filters Using Least-Squares Phase Error Criterion, *IEEE Transactions on Circuits and Systems–II: Analog and Digital Signal Processing*, **41** No. 1 (1994), pp. 40–48.

[26] LAAKSO, T. I. – VÄLIMÄKI, V. – KARJALAINEN, M. – LAINE, U. K., Splitting the Unit Delay—Tools for Fractional Delay Filter Design, *IEEE Signal Processing Magazine*, **13** No. 1 (1996), pp. 30–60.

[27] WEINREICH, G., Coupled Piano Strings, *J. Acoust. Soc. Am.*, **62** No. 6 (1977), pp. 1474–1484.

[28] ARAMAKI, M. – BENSA, J. – DAUDET, L. – GUILLEMAIN, P. – KRONLAND-MARTINET, R., Resynthesis of Coupled Piano String Vibrations Based on Physical Modeling, *Journal of New Music Research*, **30** No. 3 (2001), pp. 213–226.

[29] BANK, B., Accurate and Efficient Modelling of Beating and Two-Stage Decay for String Instrument Synthesis, In: *Proc. MOSART Workshop on Curr. Res. Dir. in Computer Music*, Barcelona, Spain, November 2001, pp. 134–137.

[30] VÄLIMÄKI, V. – KARJALAINEN, M. – LAAKSO, T., Modeling of Woodwind Bores with Finger Holes, In: *Proc. of the International Computer Music Conference (ICMC'93)*, Tokyo, Japan, 10–15 September 1993, pp. 32–39.

[31] KARJALAINEN, M. – VÄLIMÄKI, V. – RÄISÄNEN, H. – SAASTAMOINEN, H., DSP Equalization of Electret Film Pickup for Acoustic Guitar, *Proc. 106th AES Conv., Preprint No. 4907*, Munich, Germany, 1999.

[32] SMITH, J. O., Efficient Synthesis of Stringed Musical Instruments, *Proc. Int. Computer Music Conf.*, Tokyo, Japan, September 1993, pp. 64–71.

[33] JOT, J.-M., An Analysis/Synthesis Approach to Real-Time Artificial Reverberation, In: *Proc. IEEE Int. Conf. on Acoust., Speech, and Sign. Proc.*, San Francisco, California, USA, **2** (1992), pp. 221–224.

[34] BANK, B. – DE POLI, G. – SUJBERT, L., A Multi-rate Approach to Instrument Body Modelling for Real Time Sound Synthesis Applications, In: *Proc. 112th AES Conv.*, Munich, Germany, May 2002, Preprint No. 5526.

[35] BOUTILLON, X., Model for Piano Hammers: Experimental Determination and Digital Simulation, *J. Acoust. Soc. Am.*, **83** No. 2 (1988), pp. 746–754.

[36] BORIN, G. – DE POLI, G. – ROCCHESSO, D., Elimination of Delay-Free Loops in Discrete-Time Models of Nonlinear Acoustic Systems, *IEEE Trans. on Speech and Aud. Proc.*, **8** No. 5 (2000), pp. 597–606.

[37] BANK, B., Nonlinear Interaction in the Digital Waveguide with the Application to Piano Sound Synthesis, In: *Proc. Int. Computer Music Conf.*, Berlin, Germany, 2000, pp. 54–58.

[38] SMITH, J. O., Efficient Simulation of the Reed-Bore and Bow-String Mechanisms, In: *Proc. of the International Computer Music Conference (ICMC'86)*, Den Haag, Netherlands, October 1986, pp. 275–280.

[39] TAKALA, T. – HIIPAKKA, J. – LAURSON, M. – VÄLIMÄKI, V., An Expressive Synthesis Model for Bowed String Instruments, *Proc. of the International Computer Music Conference (ICMC'2000)*, Berlin, Germany, 27 August – 1 September 2000, pp. 70–73.