COMBINING DIGITAL WAVEGUIDE AND FUNCTIONAL TRANSFORMATION METHODS FOR PHYSICAL MODELING OF MUSICAL INSTRUMENTS

LUTZ TRAUTMANN[1], BALÁZS BANK[2], VESA VÄLIMÄKI[3,4], AND RUDOLF RABENSTEIN[1]

[1] *University of Erlangen-Nuremberg, Telecommunications Laboratory,*
Cauerstr. 7, D-91058 Erlangen, Germany
traut@LNT.de, rabe@LNT.de
[2] *Budapest University of Technology and Economics, Dept. of Measurement and Information Systems,*
H-1521 Budapest, Hungary
bank@mit.bme.hu
[3] *Helsinki University of Technology, Laboratory of Acoustics and Audio Signal Processing,*
P.O. Box 3000, FIN-02015 HUT, Espoo, Finland
[4] *Tampere University of Technology, Pori School of Technology and Economics,*
P.O. Box 300, FIN-28101 Pori, Finland
Vesa.Valimaki@hut.fi

Digital sound synthesis based on physical models is realized in real-time applications mostly with the well known digital waveguide method (DWG). It approximates the underlying physical behavior of a vibrating structure in a computationally efficient way. Due to these computational efficient approximations, the waveguide method looses the direct connection to the parameters of the underlying physical model. The recently introduced functional transformation method (FTM) on the other hand solves the underlying physical model analytically. Thus, the physical parameters are explicitly given in the discrete realization of the FTM. But due to this 'physicality' the computational cost of synthesis using FTM is larger than using DWG. This paper compares the DWG with the FTM and shows that for linear vibrating strings it is always possible to design an acoustically indistinguishable DWG approximation with the parameters obtained from the FTM. In that way, a computationally efficient and physically meaningful synthesis method is obtained. Furthermore, this paper shows the limits of this new synthesis method.

INTRODUCTION

Digital sound synthesis based on physical modelling has gained significant interest in the last two decades [1, 2, 3, 4]. By modelling the sound production mechanisms, traditional instruments can be understood and reproduced more intuitively than with classical signal based methods. With physical modelling methods the musician can, e.g., intuitively change the length of a string to obtain another pitch value [1, 2] or vary the string's tension for vibrato [5]. Furthermore, it is possible to control parameters of the physical model that cannot be changed intuitively in the real-world instrument, for example the geometry or the material of a string [6]. Thus, with physical modelling it is not only possible to imitate traditional instruments but also to generate new instruments with physically meaningful control parameters never used before.

For the construction of such a physical modelling algorithm a two-step procedure is necessary [6, 7].

1. Find a mathematical description of a physical model that incorporates the dominating effects of the real-world instrument.
2. Derive a computational efficient algorithm that allows real-time simulations of the mathematical description.

With regard to the second condition, only the dominating effects of the main vibrating structure, like the audible string vibration of a guitar, are considered for the mathematical description. All other insignificant physical behaviors are neglected and the surrounding structures like the guitar body are treated by fixed filters [3]. The analysis of this main vibrating structure using basic physical laws leads to multidimensional (MD) models in form of partial differential equations (PDEs) [7]. These PDEs include temporal and spatial derivatives, initial and boundary conditions as well as excitation functions. With the derivation of initial-boundary value problems in form of PDEs the first step for the design of a physical modelling algorithm, the mathematical description, is done.

The second step is the efficient implementation of the mathematical model. Since it contains temporal and spatial derivatives, it cannot be implemented in the computer directly. For discrete simulations, the initial-boundary value problem must be discretized with respect to time and space. This can be easily done by finite difference methods, where the temporal and spatial derivatives are replaced by difference terms [7, 8]. This procedure results in a set of difference equations that can directly be implemented in the computer. The drawback of this method is that for stable and accurate solutions the spatial as well as the temporal step size are restricted to small values [7]. Therefore, many points on the vibrating structure have to be computed at small time intervals, which is computationally costly. Thus, the finite difference method is not practical for real-time applications. For that purpose different methods have been used. Two of them are discussed in detail here: the digital waveguide method (DWG) and the functional transformation method (FTM).

The DWG is the most popular physical modelling sound synthesis method because of its simplicity and computational efficiency [2, 3, 9]. It analyzes the structure of the PDE qualitatively and develops efficient algorithms for the approximation of the audible results. The coefficients needed for the simulation are extracted from the measured sound of a musical instrument [3, 5]. Since the coefficients are not calculated directly from the physical model parameters, they are not physically meaningful. Thus, quantitative changes in the physical model parameters cannot be made intuitively with the DWG.

The FTM, on the other hand, calculates the analytical solutions of the underlying physical models, given in form of PDEs [6, 10, 11]. The FTM calculates the eigenmodes and their weights, depending on the physical parameters of the vibrating structure and the excitation signal. Since an analytical solution is obtained, the filter coefficients depend directly on the physical parameters of the vibrating structure. This allows intuitive sound variations just by adjusting the physical parameters. Although the vibration and parameter updates based on physical model changes can be calculated in real time [6], the FTM requires more computation power than the DWG.

To figure out the principal difference between the two methods, the derivation of the coefficients for both synthesis methods is shown on the example of a guitar string. If we want to model a guitar with nylon strings, the DWG must analyze its sound to extract its model parameters. For a change from nylon to steel strings, the real-world instrument must be changed, so that its sound can be analyzed for the DWG. The FTM on the other hand gets its model parameters from known physical string properties, such as the string's stiffness. Therefore, a more accurate physical model is needed for the FTM in comparison to the DWG. However, once this model is found, the change from nylon to steel strings only needs an adjustment of the physical parameters in the virtual model.

The natural way to combine these two methods for digital sound synthesis is to use the advantages of both methods while their disadvantages are omitted. We start with the derivation of the analytical solution of the underlying PDE by using the FTM. With that solution we design the different filters for the sound simulation with the DWG. This procedure guarantees a physically meaningful sound simulation with a computationally efficient realization. This algorithm is presented here for the example of a transversal vibrating string.

The paper is organized as follows: Section 1 describes the underlying physical model of a vibrating dispersive and lossy string. In section 2 both methods, the DWG and the FTM, are reviewed briefly. Section 3 combines the DWG and the FTM to a physically meaningful and computationally efficient method. Section 4 shows the limits of this new algorithm.

1. PHYSICAL MODEL OF STRINGS

In this section we present the physical model of a vibrating string in form of a PDE. For the solution of this PDE we also need boundary and initial conditions. To keep the example simple, we assume a linear behavior and a homogeneous material [7].

The physical model can be derived from the basic laws of elasticity. It is formulated as a PDE for the transversal deflection $y(x,t)$ of the vibrating string. No derivations are given for the following PDE, since it is well documented in the literature [7, 8]. We consider a string of length l, cross section area A and the moment of inertia I. The string material is characterized by its density ρ and its Young's modulus E. The tension T_F is applied to the string and d_1 and d_3 are the frequency independent and frequency dependent decay variables, respectively. The string can be deflected by the force density $f(x,t)$. With space and time coordinates x and t, we obtain

$$\rho A\ddot{y} - T_\mathrm{F}y'' + EIy'''' + d_1\dot{y} + d_3\dot{y}'' = f\,, \qquad (1)$$

where prime denotes spatial derivative and dot denotes temporal derivative. Space and time dependencies are omitted for concise notation. The first two terms in (1) denote the one-dimensional wave equation. These are the dominant terms of most strings in musical instruments. The one-dimensional wave equation is well known for its ideally harmonic spectrum. The third term denotes dispersion due to string stiffness, causing a stretched line spectrum in comparison to the harmonic one [12]. The last two terms characterize the damping as a result of laminar air damping and the viscoelasticity of the string material, respectively. They result in an exponential decay of the single partials.

At the nut end ($x = 0$) the string is assumed to be fixed, so that the deflection and the skewness at this point are zero [7]. This leads to homogeneous boundary conditions of first kind

$$y(0,t) = 0, \quad y''(0,t) = 0\,. \qquad (2)$$

At the bridge side ($x = l$) energy is transferred to the resonant body which amplifies and also colors the string vibration for sound radiation. The energy transfer from string vibrations to radiated sound results in a damping of the string vibration at the bridge, whereas the filtering function of the bridge and the resonant body changes the modes of the string vibration smoothly [8, 12].
Neglecting the damping terms with coefficients d_1 and d_3 in (1) at the bridge position, the boundary condition for the string can be formulated with a temporal operator D containing a linear combination of temporal derivatives by

$$\frac{f_\varepsilon(l,t)}{v(l,t)} = \mathrm{D}\left\{r_l(t)\right\}, \quad y''(l,t) = 0\,. \qquad (3)$$

These are non-homogeneous boundary conditions of third kind. $f_\varepsilon(l,t) = T_\mathrm{F} y'(l,t) - EIy'''(l,t)$ is the vibrational force of the string on the bridge and $v(l,t) = \dot{y}(l,t)$ is the transversal velocity of the string at $x = l$, respectively. For a pure resistive fixing of the string at the bridge, the right hand side of (3) results in a constant r_l. For other kinds of boundary functions $\mathrm{D}\left\{r_l(t)\right\}$ it results in a frequency dependent filter.
Since the excitation of the string shall be caused by the force density $f(x,t)$ only, we consider homogeneous initial conditions for the deflection $y(x,0)$ and the velocity $\dot{y}(x,0)$ of the string.

$$y(x,0) = 0, \quad \dot{y}(x,0) = 0, \qquad t = 0\,. \qquad (4)$$

Equations (1-4) define the initial-boundary value problem describing the vibrations of the dispersive, lossy string. Due to the temporal and spatial derivatives, this problem cannot be solved directly in the computer. Instead, discrete methods approximating the continuous solution have to be found. Two of them are presented in the next section.

2. OVERVIEW OF TWO PHYSICAL MODELLING SYNTHESIS METHODS

In this section an overview of two physical modelling methods, the DWG and the FTM, is given. First, both methods are described separately, then connections and differences between their realization structures are discussed.

2.1. Digital Waveguide Method

The basic DWG model of a stringed instrument consists of three parts [13, 14]: 1) an excitation wavetable, modelling the plucking or striking of the string, 2) the delay line loop, modelling the string vibrations, and 3) a filter for resonant body simulations. This is shown in figure 1. In linear models, the body filtering can also be incorporated into the excitation functions, well known as commuted waveguide synthesis [13, 15].

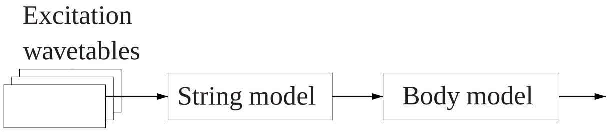

Figure 1: Basic DWG stringed instrument model consisting of an excitation wavetable, the delay line loop and the body filter.

The single components of the DWG are reviewed in detail now. The most complicated part is the simulation of the string vibrations, given in (1). To obtain an efficient implementation, the DWG simplifies in a first step the PDE to the one-dimensional wave equation. It has an analytical solution with a forward and a backward travelling wave, both with velocity $c = \sqrt{T_\mathrm{F}/(\rho A)}$. This is called the d'Alembert solution of the wave equation. After discretization with respect to time and space the d'Alembert solution can be implemented efficiently by delay lines [2]. The relation between spatial step size h and temporal step size T must be chosen to equal the wave velocity, $c = h/T$. Since in most applications the sampling rate $f_\mathrm{s} = 1/T$ is given (e.g. 44.1 kHz), the spatial step size has to be adjusted. When rigidly terminating this string at both ends, the delay line loop produces a harmonic, non-decaying spectrum [12].
In the second step, the DWG qualitatively takes further acoustical characteristics of the physical model into account. This is on the one hand the dispersion, leading to a stretched line spectrum, resulting from string stiffness and a spring-like termination at the bridge end [12]. On the other hand, the partials are exponentially decaying due to air damping, viscoelasticity of the material and the losses at the bridge [12]. Both effects are concentrated in the delay line loop with low order filters, an allpass filter for the dispersion and a loss filter for the losses [2, 13]. To adjust the model to the right pitch, a fractional delay filter has also to be included in the delay line [1, 3, 14]. This basic DWG string model is shown in figure 2a with all filters combined in $H(z)$. An efficient realization is the single delay line loop, shown in figure 2b. The integer part of the phase delay $\tau_{\mathrm{p,tot}}$ of the filter $H(z)$ is taken into account in the delay line length. In the DWG, all the coefficients of the filters, as well as the delay line length L are adapted from extracted sound parameters, like the frequencies of the partials and the temporal evolution of their amplitudes [3, 5]. With analyzed frequencies the delay line length, the dispersion filter and the fractional delay filter can be approximated in a least squares (LS) sense [16, 17]. For simplicity, all filters are assumed to have different effects on the sound signal, such that they can be designed independently from each other.
From the mathematical point of view, the string model derived above can only simulate the left hand side of (1) since it contains no excitation functions. Therefore, in

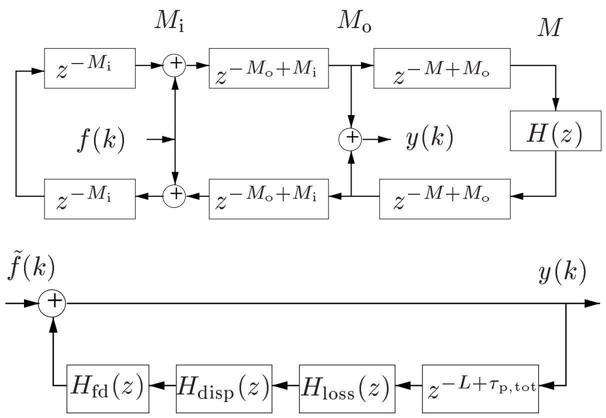

Figure 2: Basic DWG string model consisting of (a) two delay lines of length M, simulating the left and the right travelling wave, (b) the corresponding single delay line loop of length $L = 2M$. Both incorporate a dispersion filter H_disp, a loss filter H_loss and a fractional delay filter H_fd. In (a) they are combined to $H(z)$.

a third step, the excitation signal $f(k)$ is taken into account. In commuted waveguide synthesis it is filtered with the impulse response of the resonant body and stored in a wavetable. To derive this body filtered excitation wavetable, inverse filtering is applied to the recorded sound [3].

The length of the wavetable can be further reduced by serial filtering with a plucking point filter to derive $\tilde{f}(k)$ [1, 3] or by parallel filtering with the lowest body resonances [3, 14, 18]. Extensions to this fundamental DWG are also made with respect to incorporated nonlinearities [5], two polarizations in the string vibration and couplings to other strings in the instrument [1, 9]. Since only the basic models of DWG and FTM should be compared and combined in this paper, these extensions are not described here.

It has been shown that the DWG adjusts its filter coefficients as well as the excitation signal to recorded signals. Furthermore, the simplification of (1) to the wave equation implicates that the filter coefficients are not directly connected to the physical parameters of (1). Thus, the DWG can be interpreted as a sound-based physical modelling method.

2.2. Functional Transformation Method

The FTM is a direct physical modelling approach rather than a sound-based one as the DWG. It solves the initial-boundary value problem (1-4) analytically and simulates this analytical solution [4, 6, 11]. For the derivation, functional transformations with respect to time and space are used.

This procedure is well known for one-dimensional systems given by ordinary differential equations (ODEs). For applications such as systems theory, electrical network theory or control theory the temporal transformation (e.g., Laplace transformation) leads to one-dimensional transfer functions. For that kind of frequency domain description, discretization schemes are available to obtain discrete models that can be implemented efficiently. The FTM extends this approach to MD systems by also applying a functional transformation to the spatial variables. This transformation is called Sturm-Liouville transformation [19]. Further properties of this transformation can be found in [10].

The application of both transformations, the temporal Laplace transformation and the spatial Sturm-Liouville transformation, on the initial-boundary value problem (1-4) leads to the MD transfer function description

$$\bar{Y}(\mu, s) = \bar{G}_f(\mu, s)\bar{F}(\mu, s)\,. \tag{5}$$

The functions $\bar{Y}(\mu, s)$ and $\bar{F}(\mu, s)$ are the deflection of the string and the excitation force density in the temporal and spatial frequency domain, respectively; s is the temporal frequency variable and μ is an integer variable of the spatial transformation [6]. With the simplification that the string is also fixed at the bridge, the MD transfer function can be written as [4]

$$\begin{aligned} \bar{G}_f(\mu, s) &= \frac{1}{\rho A\, s^2 + e_1 s + e_2}\,, & (6)\\ e_1 &= \left(d_1 - d_3(\mu\pi/l)^2\right)\,, & (7)\\ e_2 &= EI(\mu\pi/l)^4 + T(\mu\pi/l)^2\,. & (8) \end{aligned}$$

For boundary conditions of the third kind at the bridge (4), the transfer function can also be calculated, but it is more complicated than for the case of first-order boundary conditions. Therefore it is not described here. Derivations for the pure resistive third-order boundary condition can be found in [20]. It can be seen that the MD transfer function (6) only depends on the temporal frequency variable s and the spatial integer variable μ and on the physical parameters of the string (E, I, ρ, A, l, T_F, d_1, and d_3). Inverse Laplace and Sturm-Liouville transformation lead to the continuous-time and -space solution

$$y(x, t) = \frac{2}{l} \sum_{\mu=-\infty}^{\infty} \bar{y}(\mu, t) \sin\left(\frac{\mu\pi}{l} x\right)\,. \tag{9}$$

With $*$ denoting temporal convolution, the deflection in the spatial frequency domain $\bar{y}(\mu, t)$ is given by

$$\bar{y}(\mu, t) = \frac{1}{\rho A \omega_\mu} e^{-\sigma_\mu t} \sin\left(\omega_\mu t\right) * \bar{f}(\mu, t)\,. \tag{10}$$

The important angular frequencies ω_μ as well as the decay rates σ_μ are obtained analytically ($\mu = 1$ indicates the fundamental frequency) with

$$\omega_\mu^2 \;=\; \left[\frac{EI}{\rho A} - \frac{d_3}{(2\rho A)^2}\right] \left(\frac{\mu\pi}{l}\right)^4 +$$

$$+\left[\frac{T_{\mathrm{F}}}{\rho A}+\frac{d_1 d_3}{2(\rho A)^2}\right]\left(\frac{\mu\pi}{l}\right)^2-\frac{d_1^2}{(2\rho A)^2} \quad (11)$$

$$\sigma_\mu = \frac{d_1 - d_3(\mu\pi/l)^2}{2\rho A}. \quad (12)$$

The solution of the FTM (9,10) is an extension of the Bernoulli solution, since it calculates the eigenfrequencies of the string [12]. Discretization of this continuous-time and -space system leads to an implementable system. Sampling $y(x,t)$ from (9) at $t = kT$ with temporal step size T and discrete time variable k and applying the impulse invariant transform leads to a parallel arrangement of second-order resonators, shown in figure 3. Note that the number of parallel recursive systems must be limited such that aliasing is negligible. Since all audible frequencies are exactly reproduced with a sufficiently high sampling rate (e.g. 44.1 kHz), there is no audible difference to the continuous model (9).

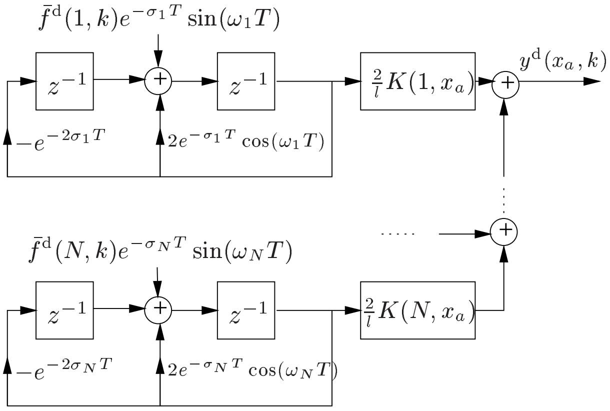
$\bar{f}^{\mathrm{d}}(1,k)e^{-\sigma_1 T}\sin(\omega_1 T)$
z^{-1}
z^{-1}
$\frac{2}{l}K(1,x_a)$
$y^{\mathrm{d}}(x_a,k)$
$-e^{-2\sigma_1 T}$
$2e^{-\sigma_1 T}\cos(\omega_1 T)$
$\bar{f}^{\mathrm{d}}(N,k)e^{-\sigma_N T}\sin(\omega_N T)$
z^{-1}
z^{-1}
$\frac{2}{l}K(N,x_a)$
$-e^{-2\sigma_N T}$
$2e^{-\sigma_N T}\cos(\omega_N T)$

Figure 3: Basic structure of the linear FTM with N second-order resonators in parallel.

The body resonances can be incorporated into this model by additional weighting of the single resonator outputs by the corresponding magnitude response of the body at these frequencies. For the lowest body resonances, also additional parallel recursive systems (with fixed coefficients) can be used. For this model-based approach of the body simulation, it is necessary to measure the magnitude response of the guitar body or to calculate it from a body model with numerical methods. With this simplification the transient response of the body cannot be taken into account, which leads to audible differences in the attack of the sound. Similarly to commuted waveguide synthesis it is also possible in the FTM to commute the output position weighting $K(\mu,x_a)$ as well as the body resonance weighting into the input position weighting $e^{-\sigma_\mu T}\sin(\omega_\mu T)$ to make the simulation more efficient.

It has been shown in this section that the FTM solves the physical model in form of the PDE analytically. Thus, parameter changes in the physical model have direct effect on the simulation results and the FTM can be called a direct physical modelling approach. The relations between the sound-based physical modelling approach of the DWG and the direct physical modelling approach of the FTM are shown in the next section.

2.3. Comparison between DWG and FTM

Although the derivations of the DWG and the FTM are quite different, several relations between the two realization structures can be pointed out. For non-dispersive and lossless strings, fixed ideally at the boundaries, the DWG realization corresponds to the time-based d'Alembert solution with a forward and backward travelling wave whereas the FTM corresponds to the frequency-based Bernoulli solution, denoting the modes of the vibration. It is well known that the continuous Bernoulli solution can be converted analytically to the continuous d'Alembert solution since both are derived from the 1-D wave equation with homogeneous first-order boundary conditions.

Discretization of the Bernoulli solution results in a parallel arrangement of $N = \lfloor L/2 - 1 \rfloor$ second-order resonators. In contrast to this, the d'Alembert solution is in discretized form one L-th order recursive delay line. The total filter order of the Bernoulli model is lower than the DWG model by 2 because the FTM do not have poles at DC and Nyquist frequencies. Due to these realizations, the simulated frequencies for the Bernoulli solution depend only on the accuracy of the coefficients (limited by the word length of the memory), whereas the frequencies derived with the d'Alembert solution depend on the total delay line length L. If the sampling frequency f_{s} is an integer multiple of the desired simulated fundamental frequency, also the discrete models of DWG and FTM can be converted into each other by simple parallel-to-serial or serial-to-parallel rearrangements. For all other fundamental frequencies, the discrete d'Alembert solution only approximates the desired modes. The required fractional part of L must be implemeted with a fractional delay filter [1, 14, 17], adjusting at least the lowest harmonics to the right pitch.

Furthermore, the spatial excitation and output positions on the string can be adjusted in the discretized Bernoulli solution continuously (discretized only with respect to the memory's word length), whereas only $M = L/2$ equally-spaced positions can be chosen for even L in the discrete d'Alembert realization without additional fractional delay filters (see figure 2a). For odd delay line lengths L, at least one fractional delay filter for each desired position is mandatory [17].

Extending the physical model from the non-dispersive and ideally fixed string to a real one with dispersion, losses and a non-ideal termination, further differences between the DWG and the FTM can be found. The DWG combines all losses and dispersion at the bridge side, whereas the basic FTM as described above, distributes all losses and dispersion effects continuously onto the string [1]. Depending on the losses and dispersion in the string of the musical instrument and on the bridge impedance, either the basic FTM or the DWG simulates the underlying model more accurate. On the other hand, the DWG can only adjust the losses and the dispersion of the lowest harmonics when low-order filters are used, whereas in case of FTM modelling the accuracy of the losses and dispersion of all partials depend only on the memory's word length.

Another difference between DWG and FTM is the increase in simulation accuracy. In the DWG the used filter orders must be stepped up to adjust more partials to the right pitch and the desired decay times. The total number of simulated frequencies stays always the same. In the FTM on the other hand, the simulated frequencies have always the right pitch and decay rates. To increase the simulation accuracy in the FTM, more partials have to be simulated.

Besides all the disadvantages of the DWG discussed above, one important advantage makes the DWG useful for real-time applications: the computational cost. A comparison between the computational cost of DWG synthesis and FTM synthesis for a nylon guitar string, further discussed in section 3, is shown in table 1. The first line of table 1 shows the number of simulated partials. Since in DWG all partials are simulated up to Nyquist frequency, these are constantly 76 partials in this example. In FTM the number of simulated partials denotes the sound quality. It can vary between 30 for low quality (LQ) and 59 for high quality (HQ), such that all partials up to Nyquist frequency are simulated. The higher number of partials in DWG compared to FTM results from the fact that the DWG only stretches the spectrum for the low partials, but not for high frequencies. In DWG the filter orders of the loss filter, the dispersion allpass and the fractional delay allpass determine the quality of simulated sound. These orders are given on the second row of table 1. For low quality, filters of first order are sufficient. For high quality and dispersive, lossy sounds, higher-order filters must be used. However, the human ear seems to be more sensitive to the number of simulated partials than to inaccuracies in the partial frequencies and decay times.

	DWG		**FTM**	
	LQ	HQ	LQ	HQ
no. of partials	76	76	30	59
DWG filter orders	1/4/1	5/10/3	-	-
FLOPS/sample	16	40	90	177

Table 1: Computational cost of DWG and FTM simulations for a nylon guitar string with fundamental frequency $f_1 = 247$ Hz, simulated at $f_s = 44.1$ kHz.

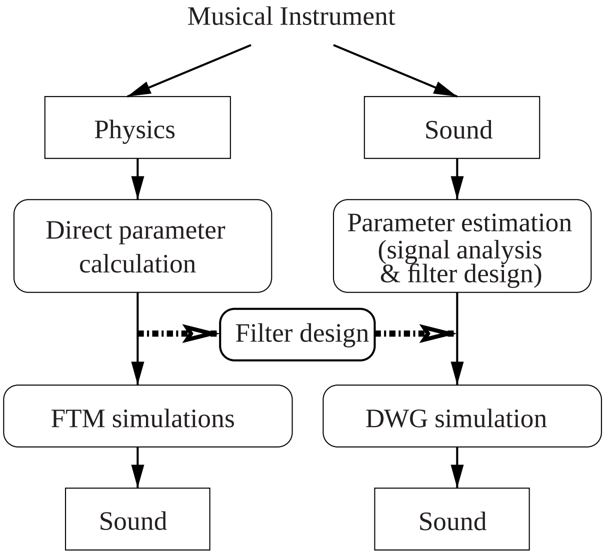
Musical Instrument
Physics
Sound
Direct parameter calculation
Parameter estimation (signal analysis & filter design)
Filter design
FTM simulations
DWG simulation
Sound
Sound

Figure 4: Combination of FTM and DWG by calculating the physics-based parameters with FTM and implementing the sound simulation with DWG.

The last line in table 1 denotes the floating point operations (FLOPS) per sample output. If we consider an IIR loss filter of 5th order, a 10th order allpass dispersion filter and a fractional delay allpass of 3rd order, the total filter order in the feedback loop of the DWG is 18. Thus, 36 multiplications and 36 additions have to be performed for the filter realization. The delay line itself can be implemented by pointer updates without mentionable computational cost [2, 15]. Adding the computational cost for the excitation (plucking point filter and output point filter), the computational amount is approximately 40 FLOPS per output sample. The FTM on the other hand needs at least three multiplications and three additions per output sample and recursive system. Thus, for the simulation of 59 partials the FTM needs 177 FLOPS per sample output, about 4.4 times more than the HQ DWG. For low quality simulations this relation even becomes 5.6.

This increase in computational efficiency lead us to the attempt to combine the direct physical modelling approach, the FTM, with the sound-based physical modelling DWG. This is discussed in the next section.

3. COMBINATION OF FTM AND DWG

The combination of FTM and DWG is depicted in figure 4. The left column shows the flow graph of the FTM, starting at the physics of a musical instrument. The DWG is depicted on the right column, starting at the sound of the musical instrument. The dash-dotted path denotes the combination of the FTM with the DWG. First, the frequencies ω_μ and the decay rates σ_μ are derived analyti-

[1] As mentioned above, the FTM has also been extended to model losses at the boundaries, but in this paper only the basic FTM is discussed.

cally from the physics of the musical instrument with the FTM. From these parameters, the different DWG filters, the loss filter H_{loss}, the dispersion filter H_{disp}, and the fractional delay filter H_{fd} are designed.

A low-order loss filter having a nearly linear or zero phase response can be designed independently from the dispersion and fractional delay filters, both affecting only the phase delay. Since both the dispersion filter and the fractional delay filter change the phase delay of the delay line loop, they cannot be interpreted to be independent from each other. However, since the dispersion filter minimizes the curve error of the delay and the fractional delay filter adjusts the absolute delay, both filters can be designed independently from each other.

The filter design methods are shown in the following sections on the example of a transversal vibrating nylon 'B' guitar string (fundamental frequency is 247 Hz), modelled with (1). The string is cylindrical with a cross section area $A = 0.5188\,\text{mm}^2$ and of length $l = 0.65\,\text{m}$. The moment of inertia is then $I = 0.171\,\text{mm}^4$. The string is made of nylon with $\rho = 1140\,\text{kg/m}^3$ and $E = 5.4\,\text{GPa}$. It has a frequency independent loss term of $d_1 = 8 \cdot 10^{-6}\,\text{kg/(ms)}$ and a frequency dependent loss term of $d_3 = -6.4 \cdot 10^{-6}\,\text{kg m/s}$. The tension of the string is $T_{\text{F}} = 60.97\,\text{N}$.

3.1. Designing the Loss Filter

The desired decay rates σ_μ of the mode frequencies ω_μ (see (10)) of the string vibration are given in analytical form by (12). It indicates that they consist of a frequency independent part d_1 and a frequency dependent part d_3, quadratically raising the decay rates of the higher frequencies. Thus, the losses have a lowpass characteristic. The DWG loss filter, combining the string damping to one low-order lowpass filter, must approximate the magnitude response

$$\left|H_{\text{loss}}\left(e^{j\omega_\mu T}\right)\right| \approx e^{-\frac{2\pi\mu\sigma_\mu}{\omega_\mu}} . \tag{13}$$

Using a one-pole loss filter with transfer function

$$H_{\text{loss},1}(z) = g\frac{1+a_1}{1+a_1 z^{-1}} \tag{14}$$

where g refers to the DC gain and $-a_1$ is the pole of the filter, an analytical approximation for these coefficients can be found. This is done for a nearly harmonic line spectrum in [18] by some low-order Taylor series approximations. It results in filter coefficients given by

$$g = 1 - \frac{\pi d_1}{\rho A \omega_1}, \quad c_0 = \frac{\rho A l^2 \omega_1^3 T^2}{4\pi^3 d_3}, \tag{15}$$

$$a_1 = -1 + c_0 - \sqrt{c_0^2 - 2c_0} . \tag{16}$$

This algorithm works quite well for decay rates of normal strings. Since the Taylor series approximations are made at zero frequency, the derived one-pole lowpass approximation is accurate at low frequencies whereas it fails for high frequencies, as shown in figure 5. Because of the short desired decay times at high frequencies, the approximation error changes the synthesized sound only negligible. On the other hand this algorithm assumes a nearly harmonic spectrum of the synthesized sound, so that it is not applicable to very dispersive sounds, such as those of vibrating bars.

For these cases, an LS algorithm can be used to design higher-order IIR loss filters. We have chosen a modified version of the LS loss filter design method proposed in [18]. It minimizes the weighted sum of the squared difference between the filter decay times $\hat{\tau}_\mu = 1/\hat{\sigma}_\mu$ and the desired decay times τ_μ at frequencies ω_μ

$$e_{\text{loss}} = \sum_{\mu=1}^{N} w_{\mu,\text{loss}} \left(\hat{\tau}_\mu - \tau_\mu\right)^2 . \tag{17}$$

The weights $w_{\mu,\text{loss}}$ are chosen to give emphasis to the slow decaying partials, realized by $w_{\mu,\text{loss}} = \sigma_\mu^2$. This procedure gives accurate results for a wide frequency range at order five, shown in figure 5.

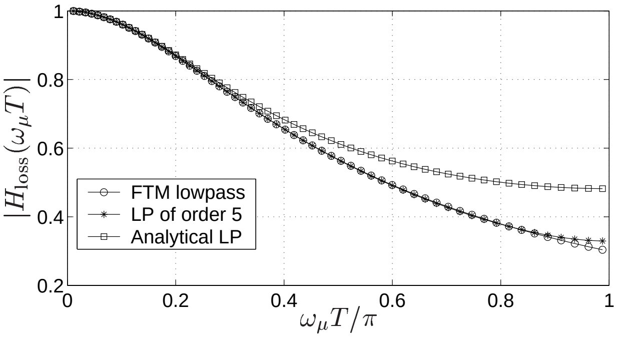

Figure 5: Magnitude response (MR) of the loss filter for a DWG model. Desired MR, derived by FTM (circles), MR of the analytical obtained one-pole filter (squares), MR of the 5th order IIR filter obtained by LS algorithm (∗).

The LS method for the loss filter design, as described above, presumes that the DWG simulates all partials at the right pitch. To derive an approximation of that, a dispersion filter of high order must be designed. This is done in the next section.

3.2. Designing the Dispersion Filter

Dispersion denotes an increase of the frequencies of the higher partials. In DWG formulation this increase results in a shorter delay line for the higher partials than for the lower ones. For the realization of such a task, a filter with a non-constant phase delay is required. Since the magnitude response of the delay line should not be affected by the dispersion filter, it is straightforward to use an allpass filter. For the design, we use the method presented in [21, 22].

The desired phase delay $\tau_{\mathrm{p,FTM}}$ is given by

$$\tau_{\mathrm{p,FTM}}(\omega_\mu T) = \frac{2\pi\mu}{\omega_\mu T} - \tau_{\mathrm{p,loss}}(\omega_\mu T)\,. \qquad (18)$$

It takes the phase delay $\tau_{\mathrm{p,loss}}$ of the already designed loss filter into account. For dispersive strings or thin bars, $\tau_{\mathrm{p,FTM}}$ is a continuously decreasing function. The designed method proposed in [22] minimizes the weighted squared phase error between the desired phase response $\Phi_{\mathrm{FTM}} = \omega_\mu \tau_{\mathrm{p,FTM}}$ and the phase response of the dispersion filter Φ_{disp}. First we compute the quantities

$$\kappa_\mu = -\frac{1}{2}(\Phi_{\mathrm{FTM}}(\omega_\mu) + P\omega_\mu) \qquad (19)$$

and solve (20) for the coefficients a_k of the filter denominator (with $a_0 = 1$),

$$\sum_{k=1}^{P} a_k \sin(\kappa_\mu + k\omega_\mu) = -\sin\kappa_\mu, \quad \mu = 1, \ldots, N\,. \qquad (20)$$

Since the number of prescribed phase values N is higher than the filter order P, the set of equations is overdetermined, thus it cannot be precisely solved. However, when the equation error is minimized in the mean-squares sense, the solution is easily computed.

The error analysis shows that the phase error is weighted by the magnitude response of the denominator of the allpass filter $|D_{\mathrm{AP}}(e^{j\omega T})|$ when the allpass filter is designed by minimizing the squared sum of the equation error. If the squared inverse of this magnitude response is used as a weighting function $w_{\mu,\mathrm{comp}} = |D_{\mathrm{AP}}(e^{j\omega_\mu T})|^{-2}$ when computing the LS solution, the mean-squared error of the phase response is minimized. Since $|D_{\mathrm{AP}}(e^{j\omega T})|$ is not known beforehand, the weighted LS solution has to be computed iteratively with a weighting function computed from the allpass filter of the previous iteration $w^{(q)}_{\mu,\mathrm{comp}} = |D^{(q-1)}_{\mathrm{AP}}(e^{j\omega_\mu T})|$, where q is the sequential number of the iteration.

An additional weighting function is used to minimize with the algorithm described above the phase delay error instead of the phase error. Therefore, the weights take the form $w_{\mu,\mathrm{disp}} = w_{\mu,\mathrm{comp}}/\omega_\mu^2$.

For higher filter orders the condition number of equation (20) increases significantly, therefore it is advisable to use several lower order allpass filters in series. If the order and the specification of these filters are the same, the design procedure remains the same, the only difference is that we use $\tilde{\Phi}_{\mathrm{FTM}} = \Phi_{\mathrm{FTM}}/K_{\mathrm{disp}}$ as the phase specification in (19), where K_{disp} is the number of allpass filters in series. Filter orders of 4 to 6 have been found good in practice.

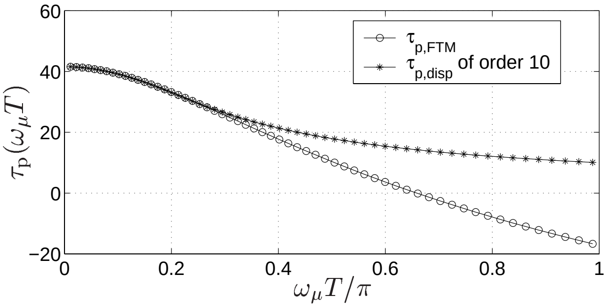

Figure 6: Phase delay of the dispersion filter. Desired phase delay derived by FTM (circles) and phase delay of two allpass filters of 5th order obtained by LS algorithm (∗).

By perceptual reasons, it is not necessary to match all the partial frequencies ω_μ of the FTM perfectly by the DWG simulation [16]. Therefore, the number of specification points N in (20) usually does not need to exceed 20 or 30, and an additional weighting function is used to put larger emphasis on the lower partials. Results for the nylon string are shown in figure 6 for two 5th order allpass filters in series.

To tune the single delay line loop of the DWG to the right pitch the delay line length has to be adjusted. It can be subdivided into an integer part $L - \tau_{\mathrm{p,tot}}$ (see figure 2b), realized by a delay line and a fractional part, realized by a fractional delay filter $H_{\mathrm{fd}}(z)$. The fractional delay filter is realized with a third order Thiran allpass filter that does not change the magnitude response. This filter introduces, besides the desired constant phase delay for low frequencies, a negligible phase delay error for higher partials (see e.g. [17]).

After calculating the loss, dispersion and fractional delay filters, the string model for the DWG is designed. In the next section, the excitation signal applied to the string derived with FTM is adjusted for the DWG simulation.

3.3. Adjusting the Excitation Function

With the single delay line loop, completed by the filters designed in sections 3.1 and 3.2, the DWG model approximates the frequencies ω_μ and the decay rates σ_μ, given by the FTM. To excite the single frequencies in the DWG in the same way as the FTM does, the excitation signal from the FTM has to be adjusted to the single delay line loop of the DWG.

As known from commuted waveguide synthesis [15, 13], the excitation function $f(x,t)$ has to be convolved with $H_{\mathrm{in}} = 1 - z^{-2M_{\mathrm{i}}}$ (see figure 2a). In the same way the output position must be filtered, such that $H_{\mathrm{out}} = 1 - z^{-(L-2M_{\mathrm{o}})}$. The excitation signal should also be delayed by $z^{-(M_{\mathrm{o}}-M_{\mathrm{i}})}$. Both the excitation point and the output point inclusion into the excitation wavetable as stated above assume a lossless and non-dispersive string

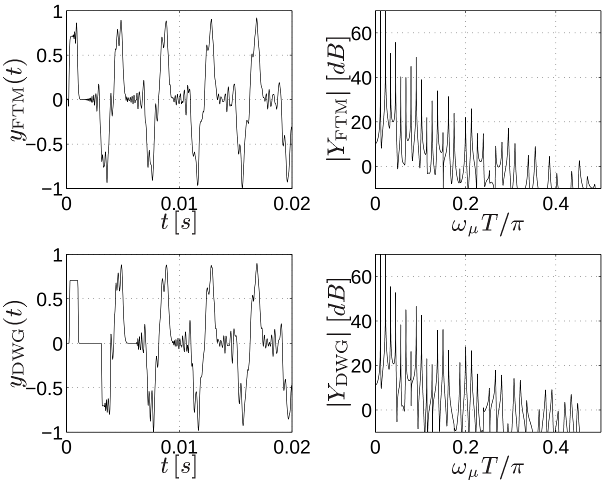

Figure 7: Deflection and spectrum of the example nylon guitar string, simulated with FTM (upper plot) and with DWG (lower plot).

behavior as well as ideal first-order boundary conditions. Since the combination of FTM and DWG should also take lossy and dispersive strings into account, further adjustments have to be done.

On one hand, the delayed excitation signal will be damped and on the other hand it will be dispersed, such that the high frequency components of the excitation signal arrive earlier than the lower ones. These effects can be included in the excitation wavetable of the DWG by filtering the delayed excitation signals with low-order filters similar to the filters designed for the delay line loop. These filters do not have to be simulated as accurately as the filters for the delay line loop. We chose to design the one-pole loss filter after (15,16) with ω_1 replaced by $\omega_{\mathrm{exc}} = L\omega_1/L_{\mathrm{part}}$. This takes into account that the signal only travels along L_{part} delays of the whole delay line of length L.

A similar approximation can be done to design the excitation dispersion filters. The desired phase delays in these cases are given by $\tau_{\mathrm{p,disp}}L_{part}/L$ instead of $\tau_{\mathrm{p,disp}}$.

Figure 7 shows the example string deflection as well as the signal spectra at distance $x_a = 20\,\mathrm{cm}$ from the bridge position, simulated with FTM and DWG. The string is struck at the position $x_e = 12\,\mathrm{cm}$ from the bridge. It can be seen that the signal deflections look similar after the first five milliseconds. The spectra are only similar for the lowest partials due to the dispersion filter approximation. The rough adjustment of the excitation signal causes another difference between the signals. In the DWG simulation the low order dispersion filter can only disperse the excitation signal within a few time steps, e.g. at $3.5\,\mathrm{ms}$. In contrast to this, the FTM simulates the dispersion accurately from the first samples on.

Since this signal difference is only at the first few milliseconds, nearly no sound difference between the simulation methods can be perceived. Thus, the combination of FTM and DWG works well for the simulation of stringed instruments.

4. LIMITS OF THE COMBINATION

It has been shown in section 3 that it is possible to design the different filters for the DWG from the sound parameters derived from the physical model by applying the FTM. This procedure is done to save computational cost for the calculation of the string vibration. In the example above, the computational saving is more than 77% in comparison to FTM simulation, neglecting the filter design. For lower fundamental frequencies or lower-order DWG filters the computational saving can be even higher. On the other hand, the computational saving of this method decreases with the desired dispersion. Changing the string material in section 3 from nylon to steel ($\rho = 7800\,\mathrm{kg/m}^3$ and $E = 200\,\mathrm{GPa}$) lowers the computational savings with the same DWG filter orders to 70%, since only 44 partials have to be calculated with FTM. For simulations of vibrating bars with a larger cross section area the computational saving is even lower for the same DWG filter orders. In addition to that, the DWG must use a higher-order dispersion filter to approximate the desired dispersion accurately.

Taking the computational cost for the DWG filter design into account, the computational savings depend on the parameter update rate of the physical model. Depending on the implementation of the DWG filter designs, 10^5 FLOPS are needed for a filter update. Then a parameter change may occur every 730 samples ($17\,\mathrm{ms}$ for $f_{\mathrm{s}} = 44.1\,\mathrm{kHz}$) such that FTM and the combination of DWG and FTM need the same computational cost. This rate becomes lower for higher dispersions or higher fundamental frequencies, so that this combination is not any more useful for high parameter update rates. Furthermore, these parameter updates can cause audible transients in the DWG since the filter coefficients are not changing as smoothly as in the FTM.

5. CONCLUSIONS

In this paper the combination of two different physical modelling sound synthesis methods, the frequency-based FTM and the time-based DWG, was presented. This combination first calculates with the FTM sound parameters such as the partial frequencies and their decay rates from the physical model of a musical instrument. These parameters are then used for the design of low-order filters included in the delay line of the DWG. It has been shown that this procedure allows a physics-based sound simulation of linear vibrating strings with low computational cost. Limitations of this combination have been found for very dispersive vibrations and for frequent parameter changes.

ACKNOWLEDGMENTS

This work was conducted when Lutz Trautmann was a visiting researcher at the Laboratory of Acoustics and Audio Signal Processing, Helsinki University of Technology, in September 2000 and November 2001. His visit was financed by the Deutsche Akademischer Austauschdienst (DAAD) which is greatfully acknowledged.

REFERENCES

[1] D.A. Jaffe and J.O. Smith. Extensions of the Karplus-Strong plucked string algorithm. *Computer Music Journal*, 7(2):56–69, 1983.

[2] J. O. Smith. Physical modeling using digital waveguides. *Computer Music Journal*, 16(4):74–91, 1992.

[3] V. Välimäki, J. Huopaniemi, M. Karjalainen, and Z. Jánosy. Physical modeling of plucked string instruments with application to real-time sound synthesis. *J. Audio Eng. Soc.*, 44(5):331–353, 1996.

[4] L. Trautmann and R. Rabenstein. Transfer function models with nonlinear excitation functions for digital sound synthesis. In *Proc. X European Signal Processing Conference (EUSIPCO 2000)*. EURASIP, Sept. 2000.

[5] 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 *108th AES Convention*, Paris, France, February 2000. Preprint 5114.

[6] L. Trautmann and R. Rabenstein. Digital sound synthesis based on transfer function models. In *Proc. Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 83–86. IEEE, 1999.

[7] A. Chaigne and A. Askenfelt. Numerical simulations of piano strings. I. A physical model for a struck string using finite difference methods. *J. Acoust. Soc. Am.*, 95(2):1112–1118, 1994.

[8] L. Hiller and P. Ruiz. Synthesizing musical sounds by solving the wave equation for vibrating objects: Part I. *J. Audio Eng. Soc.*, 19(6):462–470, 1971.

[9] M. Karjalainen, V. Välimäki, and T. Tolonen. Plucked-string models: From the Karplus-Strong algorithm to digital waveguides and beyond. *Computer Music Journal*, 22(3):17–32, 1998.

[10] R. Rabenstein. Discrete simulation models for multidimensional systems based on functional transformations. In J.G. McWhirter and I.K. Proudler, editors, *Mathematics in Signal Processing IV*, pages 335–347. Oxford University Press, 1998.

[11] R. Rabenstein. Transfer function models for multidimensional systems with bounded spatial domains. *Mathematical and Computer Modelling of Dynamical Systems*, 5(3):259–278, 1999.

[12] N.H. Fletcher and T.D. Rossing. *The Physics of Muiscal Instruments*. Springer-Verlag, New York, 1998.

[13] M. Karjalainen, V. Välimäki, and Z. Jánosy. Towards high-quality sound synthesis of the guitar and string instruments. In *Proc. Int. Computer Music Conference*, pages 56–63, 1993.

[14] V. Välimäki and T. Tolonen. Development and calibration of a guitar syntheziser. *J. Audio Eng. Soc.*, 46(9):766–778, 1998.

[15] J. O. Smith. Efficient synthesis of stringed musical instruments. In *Proc. Int. Computer Music Conference*, pages 64–71, 1993.

[16] D. Rocchesso and F. Scalcon. Bandwidth of percieved inharmonicity for physical modeling of dispersive strings. *IEEE Trans. Speech and Audio Processing*, 7(5):597–601, Sept. 1999.

[17] T.I. Laakso, V. Välimäki, M. Karjalainen, and U.K. Laine. Splitting the unit delay — tools for fractional delay filter design. *IEEE Signal Processing Magazine*, 13(1):30–60, 1996.

[18] B. Bank. Physics-based sound synthesis of the piano. Master's thesis, Budapest University of Technology and Economics, Budapest, Hungary, May 2000. Published as report 54 of the Laboratory of Acoustics and Audio Signal Processing, Helsinki University of Technology. Available at http://www.acoustics.hut.fi/publications/.

[19] R.V. Churchill. *Operational Mathematics*. McGraw-Hill, New York, 3. edition, 1972.

[20] L. Trautmann and R. Rabenstein. Turning vector partial differential equations into multidimensional transfer function models. *Mathematical and Computer Modelling of Dynamical Systems*. in print.

[21] D. Rocchesso and F. Scalcon. Accurate dispersion simulation for piano strings. In *Proc. Nordic Acoustical Meeting*, pages 407–414, Helsinki, Finland, 1996. The Acoustical Society of Finland.

[22] M. Lang and T. I. Laakso. Simple and robust method for the design of allpass filters using least-squares phase error criterion. *IEEE Trans. Circuits and Systems–II: Analog and Digital Signal Processing*, 41(1):40–48, Jan. 1994.