

Jyri Pakarinen, Vesa Välimäki, and Matti Karjalainen, 2005, Physics-based methods for modeling nonlinear vibrating strings, *Acta Acustica united with Acustica*, volume 91, number 2, pages 312-325.

© 2005 S. Hirzel Verlag

Reprinted with permission.

Physics-Based Methods for Modeling Nonlinear Vibrating Strings

Jyri Pakarinen, Vesa Välimäki, Matti Karjalainen

Helsinki University of Technology, Laboratory of Acoustics and Audio Signal Processing, P.O. Box 3000, FI-02015 HUT, Espoo, Finland. jyri.pakarinen@hut.fi

Summary

Nonlinearity in the vibration of a string is responsible for interesting acoustical features in many plucked string instruments, resulting in a characteristic and easily recognizable tone. For this reason, synthesis models have to be capable of modeling this nonlinear behavior, when high quality results are desired. This study presents two novel physical modeling algorithms for simulating the tension modulation nonlinearity in vibrating strings in a spatially distributed manner. The first method uses fractional delay filters within a digital waveguide structure, allowing the length of the string to be modulated during run time. The second method uses a nonlinear finite difference approach, where the string state is approximated between sampling instants using similar fractional delay elements, thus allowing run-time modulation of the temporal sampling location. The magnitude of the tension modulation is evaluated from the elongation of the string at every time step in both cases. Simulation results of the two models are presented and compared. Real-time sound synthesis of the kantele, a traditional Finnish plucked-string instrument with strong effect of tension modulation, has been implemented using the nonlinear digital waveguide algorithm.

PACS no. 43.75.Gh, 43.75.Wx

1. Introduction

Interest towards physical modeling for sound synthesis purposes has been increasing during the last few years. The advantages of the physical models over traditional sound synthesis methods reside in physically meaningful model parameters which allow a natural control of the synthesis engine. There has also been a trend towards comparing and unifying the existing physical modeling methods with a focus on generating more flexible and efficient sound synthesis models [1], [2]. This paper discusses and compares two physical models for simulating the behavior of the nonlinear string: a spatially distributed nonlinear digital waveguide string recently developed by the authors [3], and a new spatially distributed nonlinear finite difference string model, created as a part of a Master's Thesis work at Helsinki University of Technology [4].

Since the purpose of physics-based modeling is to simulate the physical phenomena in the system of interest, it is of paramount importance to be familiar with the behavior of the real-world case before modeling can take place. We will start by studying some physical properties of vibrating strings in section 2, the main focus being on those, which contribute the most to the resulting sound.

Section 3 will discuss digital waveguide modeling of strings and present a spatially distributed nonlinear digi-

tal waveguide string algorithm. A synthesis model of the *kantele*, a Finnish plucked-string instrument, is presented using the nonlinear digital waveguide model in section 4. We will tackle finite difference modeling of strings in section 5, and introduce a spatially distributed nonlinear finite difference string algorithm in section 6. Simulation results and comparisons to measured data are presented in section 7, and conclusions are briefly drawn in section 8.

2. Basics of string mechanics

2.1. Linear string

Let us consider a homogeneous string, which is completely flexible, linear and lossless (i.e. the string's total energy remains constant). Such a string is called an ideal string. If we also consider the string moving only in one transversal polarization (e.g. horizontal), the motion of an ideal string can be characterized by the well-known one-dimensional (1-D) wave equation:

$$y_{tt}(t, x) = c^2 y_{xx}(t, x). \quad (1)$$

Here, $y_{tt}(t, x)$ denotes the second-order partial derivative of the string displacement in the horizontal axis with respect to the time variable t , $y_{xx}(t, x)$ denotes the second-order partial derivative of the horizontal displacement with respect to the longitudinal coordinate x , and c denotes

the transversal wave propagation velocity within the string medium. The transversal wave velocity can be written as

$$c = \sqrt{\frac{K}{\epsilon}}, \quad (2)$$

where K is the string tension and ϵ is the linear mass density of the string.

Obviously, every real string vibration decays with time. This results mainly from three damping mechanisms [5]: (1) air damping, (2) internal damping in the string medium, and (3) mechanical energy transfer through string terminations. If the losses are simply divided into frequency-dependent and frequency independent terms, the lossy 1-D wave equation can be written as [6], [7]:

$$K y_{xx}(t, x) = \epsilon y_{tt}(t, x) + d_1 y_t(t, x) - d_2 y_{xxx}(t, x). \quad (3)$$

Here, d_1 and d_2 are coefficients that simulate the frequency-independent and frequency-dependent damping, respectively. Real strings also experience stiffness, which leads to dispersion of the wave velocities and inharmonicities in the resulting sound. Since this paper focuses on the sound synthesis of strings with relatively high elasticity and small diameters, the effects of stiffness are not discussed here. More in-depth study about this topic can be found for example in [8].

2.2. Nonlinear string

Although the linearity assumption usually simplifies calculations, the vibration of real strings can be considered linear in the most coarse approximations only. More realistic string models require abandoning the linearity assumption, thus also introducing more complex formulations for string behavior.

Some interesting nonlinearities in string instruments include the hammer-string contact in piano-like instruments and the bow-string interaction in bowed string instruments. These phenomena fall out of the scope of this paper. The hammer-string nonlinearity is mainly due to the compression of the hammer felt, and it is covered in many earlier studies [9], [10], [11], [12], and [13], to name few. The bow-string nonlinearity is mainly caused by the stick-slip contact between the string and the bow. Also this interaction is covered in many studies, [14], [15], and [16] present some of them. In the following, we will study more thoroughly the tension modulation nonlinearity.

When a real string is displaced, its length, and therefore also its tension is increased. When the string returns closer to its equilibrium state, its length and tension are decreased. This mechanism, where the tension is varied due to transversal vibrations, is called *tension modulation*, and abbreviated TM.

It is easy to see that the frequency of the TM is twice the frequency of the transversal vibration, since the transversal equilibrium produces minimum tension, and both extreme displacements produce maximum tension. In many musical instruments the string termination is rigid in the longitudinal direction, so that TM cannot be effectively coupled

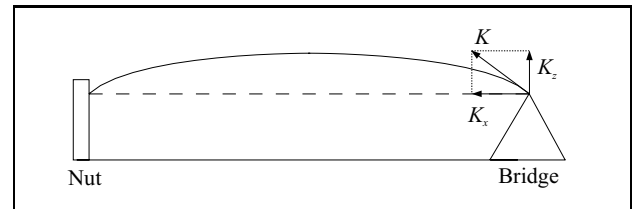


Figure 1. Tension modulation exerts a vertical force K_z on the bridge. If the bridge is able to move in the z -direction, this vibration will be coupled with the string. With a rigid bridge, no coupling will take place, and the missing harmonics will not be generated (after [19]).

to the instrument body, and therefore cannot be heard directly. In instruments where this is not the case, partials created by TM can be found at twice the frequencies of the transversal modes, and they are called *phantom partials*. The generation of these partials is discussed in more detail in [17], and in [18].

Due to decaying vibration, the string displacement is clearly greater just after the pluck than at the end of the vibration. Therefore also the amplitude envelope of TM has the same form as the transversal vibration itself. As can be seen in equation (2), the increase in tension results in the increase of wave velocities, which in turn leads to the increase in frequency. Thus, the frequency of a plucked string glides from an initial value to the steady-state value (the frequency in the linear case). This is called *initial pitch glide*, an effect which is most apparent in elastic strings with large vibrational amplitudes and relatively low nominal string tension (such as electric guitar or kantele strings).

Tension modulation is also responsible for another acoustic feature, *generation of missing harmonics*. This means, as its name implies, that the harmonics which should be missing from the vibration can be found in the spectrum. Generally, the lack of certain harmonics in a plucked string instrument occurs due to the plucking location, which heavily attenuates all harmonics that would have a node at that point. In elastic strings however, the missing harmonics begin with a gradual increase near zero amplitude until they reach their peak value, and then decay off like all other harmonics. In the following, this phenomenon is discussed, while avoiding to go too deep into the mathematics. A more in-depth study can be found from a paper by Legge and Fletcher [19].

Although the TM takes place in the longitudinal direction, it can also excite the string in the transversal directions at the string termination point, provided that the termination is nonrigid in the transversal plane. Figure 1 clarifies this. As can be seen in the illustration, the varying of the tension, called *tension modulation driving force* (TMDF) [20], excites the bridge in the vertical direction. If the bridge has a nonzero mechanical admittance in the vertical direction, TMDF will cause the bridge to move up and down.

Let us now consider a case where a vibrating string carries two transversal modes, n and m . As stated above,

the TM caused by mode n is periodic with a frequency corresponding to mode $2n$. When this vibration is coupled through the nonrigid bridge with the transversal vibration of mode m , the resulting vibration is the mode $2n$ amplitude-modulated with mode m . Thus, this vibration can excite mode p only if $p = |2n \pm m|$. Clearly, the same rule applies with m and n interchanged.

Now it is clear that even if a string is plucked at $x = L/p$, the p th mode, although supposed to be missing, will receive energy from other modes because of this nonlinear coupling at the bridge. It is important to note however, that this phenomenon cannot excite all the modes, e.g. if the string is plucked near its center so that no even modes will be present, they will not be generated by this mechanism either.

Since this energy transfer from other modes is rather gradual than instantaneous, the missing modes excited by the TMDF will experience a gradual onset, and behave like other modes after reaching their peak values. The rising rate of these missing harmonics can be shown to be proportional to the cube of the pluck amplitude [19].

In real musical instruments, also another mechanism is responsible for the generation of missing harmonics. The string is often terminated behind the bridge, and it undergoes a change in direction at the bridge location. Figure 2 illustrates this fact. Now the TM can be directly coupled with the vertical polarization, due to angle α at the bridge. This means that the TMDF due to a transversal mode n will have a frequency corresponding to mode $2n$, so this mechanism can excite only even modes, thus rising the even harmonics also in a middle-plucked string [19].

3. Digital waveguide approach

Digital waveguide (DWG) modeling is a term often encountered when studying the synthesis of string instruments. It is based on the fact that when an excitation signal is inserted into a string, it is reflected at the boundaries, and returns to its initial position. At its simplest form, this can be implemented as a single delay-loop with two consecutive samples averaged, as is done in the classic *Karplus-Strong* algorithm [21]. An excellent introduction to DWGs used in modeling musical instruments can be found in [22].

The entire digital waveguide methodology is based on the traveling-wave solution of the wave equation. This means that the solution to equation (1) can be seen as a superposition of two waveforms traveling in opposite directions along the string. This solution, commonly known as the *traveling-wave solution*, or as the *d'Alembert's solution*, was first published by d'Alembert in 1747. It can be presented in the mathematical language as [22]

$$y(t, x) = y_r(t - x/c) + y_l(t + x/c), \quad (4)$$

where y_r and y_l denote the wave components proceeding right and left, respectively. The traveling-wave solution of the 1-D wave equation (1) can be converted into digital form by sampling the wave components temporally at

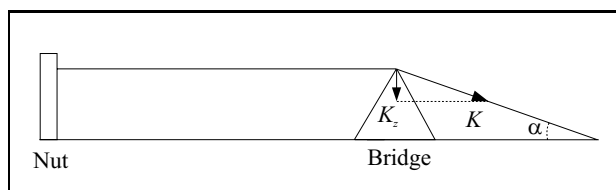


Figure 2. A more realistic bridge model. The angle α causes the tension to have a vertical component, K_z . This results in a TMDF with frequency twice as high as the transversal frequency in the string (after [19]).

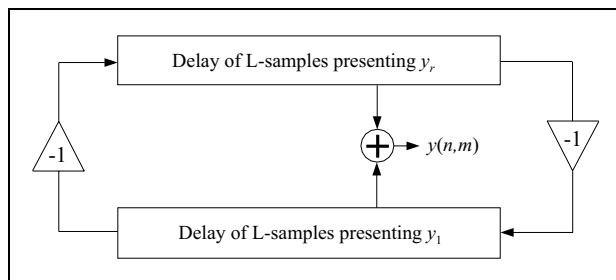


Figure 3. A DWG model of an ideal string. The wave reflection at the fixed termination points is implemented with a sign change, since $y = 0 \Rightarrow y_r = -y_l$. The string excitation can be inserted e.g. by initializing the delay lines to nonzero values (after [23]).

T and spatially at X intervals. Formally, this is done by changing the variables in equation (4) [22]

$$\begin{aligned} x &\rightarrow x(m) = mX, \\ t &\rightarrow t(n) = nT. \end{aligned}$$

If now the traveling waves are redefined as

$$\begin{aligned} y_l(n) &\leftarrow y_l(nT), \\ y_r(n) &\leftarrow y_r(nT), \end{aligned}$$

the discrete traveling-wave solution can be obtained

$$y(t(n), x(m)) = y_r(n - m) + y_l(n + m). \quad (5)$$

The term $y_r(n - m)$ can be thought as $y_r(n)$, delayed by m samples. Similarly, the term $y_l(n)$ can be thought as $y_l(n + m)$, delayed by m samples. It is important to note that equation (5) is *not* a mere approximation of equation (1), but yields *exact* results for bandlimited signals at the sampling instants, within the limits of the numerical precision of the samples [22]. This kind of structure can be easily implemented with two delay lines containing unit delays, and $y(n, m)$ can be obtained by summing the delay line values at correct locations. The string state at the next time step can be updated by simply shifting the samples one step in the direction of the delay line. A DWG model of an ideal string is shown in Figure 3.

Correct tuning of the waveguide can be enabled by adding a fractional delay filter (i.e. a filter capable of producing also noninteger delay values) inside the DWG loop [24]. Frequency-independent losses can easily be modeled in a DWG structure by inserting simple scaling coefficients in the ideal DWG string structure [22]. Frequency-

dependent losses can in turn be simulated using lowpass filters (a.k.a. *loop filters*) inside the DWG loop.

3.1. Time-varying digital waveguide string

When tension modulation is to be implemented in a string model, it clearly means that the fundamental frequency of the string must be modulated also (see equation 2). In a DWG string this corresponds to varying either the length of the delay lines or the temporal sampling instant. This section discusses implementing the TM by varying the length of the delay lines, and is presented earlier in a recent publication by the authors [3]. Since the delay lines can generally have integer-valued delays only, directly altering them would lead to having the tension change in a stepwise manner. Obviously, this behavior is not desired and therefore fractional delay (FD) elements are used. For an in-depth study of FD elements, see [24].

A first-order allpass filter was chosen for the FD element of our string model.

$$A(z) = \frac{-a + z^{-1}}{1 - az^{-1}}, \tag{6}$$

where a is the filter coefficient which defines the length of the delay. Notice that when $a = 0$, the allpass filter acts as a unit delay.

The decision for using a first-order allpass filter within the string model was done partly because it is the simplest way to design an allpass filter approximating a given fractional delay [24], and partly because the first-order allpass filter is the best choice for the fractional delay element when delay values around unity are to be obtained [25]. The phase response error caused by the allpass filter is not considered to pose a problem since its effect is negligible in the audio frequency range, assuming the sampling frequency to be reasonably high [26].

3.2. Distributed nonlinear DWG string

Previous works [27], [28], [29] use a single fractional delay element in a single-polarization string model, or a DWG string terminated with a nonlinear double-spring [30] to simulate the nonlinear string. This is done in order to reduce the computational complexity of the model, but it has also some shortcomings. Since the system is nonlinear, the FD elements cannot be lumped into one single element without giving up the idea of viewing the system as a distributed model.

In other words, the whole string becomes a lumped model, and the termination point “behind” the FD element becomes the only location for gathering meaningful output from the string. Physically this would correspond to a single elastic element at the termination point of an otherwise rigid string. A more realistic solution can be obtained if the elongation process is distributed along the delay line, in a similar way as in a real physical string, where the elasticity is distributed along the string, rather than lumped.

The distributed nonlinearity can be implemented by exchanging the delay lines of the DWG model of Figure 3

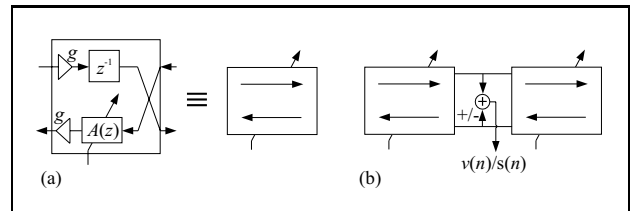


Figure 4. Illustration of (a) a basic element, and (b) how to get output data from a string consisting of these elements. The directions of the wave components in (a) are opposite for adjacent elements, so that in effect the unit delays and allpass filters are interleaved for each delay line. In (b), either the velocity or the slope of the string segment can be obtained, if velocity is used as the wave variable.

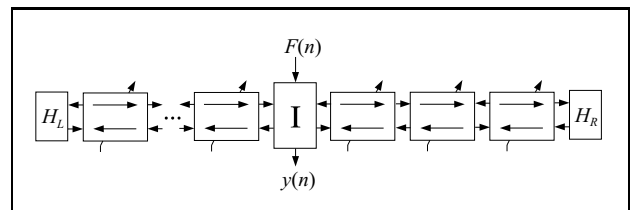


Figure 5. One-polarizational DWG string with time-varying length. The string consists of the basic elements illustrated in Figure 4(a). H_L and H_R denote the loop filters simulating the frequency-dependent losses. The excitation to the string can be inserted as a force signal using an *interaction element*, denoted by I . The construction of the interaction element is illustrated in Figure 6.

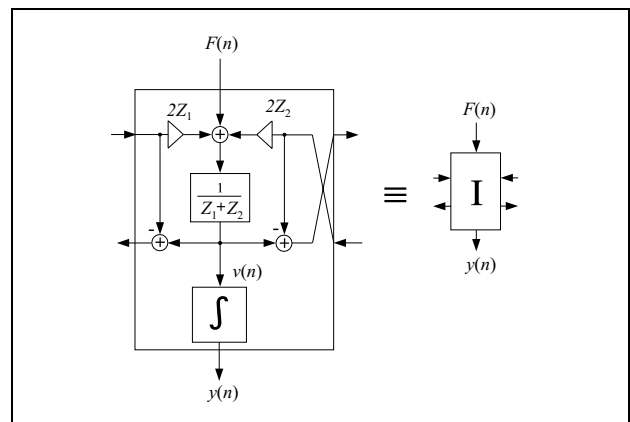


Figure 6. The interaction element allows excitation signals to be inserted to the string during run-time. The input signal $F(n)$ can be seen as a force signal, and the output signal $y(n)$ as a displacement signal. The coefficients Z_1 and Z_2 represent the mechanical impedances of the two string branches. Implementation of the integrator block is depicted in Figure 7.

with a structure consisting of allpass filters. Then the effective length of the delay lines can be changed by varying the filter coefficients. We will now introduce a delay block which contains a unit delay, a first-order allpass filter, and two scaling coefficients for modeling the frequency-independent losses. This block is called a *basic element*, and it is illustrated in Figure 4(a). The unit delay in each basic element ensures that no delay-free loops are

formed when constructing models using these elements. Figure 4(b) shows how to obtain output data from a junction between two basic elements. A time-varying DWG structure consisting of these elements is illustrated in Figure 5.

From the discussion in section 1, we can conclude that a suitable control signal for the FD elements can be derived from the instantaneous elongation of the string. In the following, since the longitudinal wave propagation velocity is considerably higher than the transversal wave velocity, we will assume that the longitudinal waves will propagate instantaneously through the string, and the elongation calculation and the FD parameter tuning can be done for the whole string in one piece. In practice, the longitudinal wave velocity is typically only 5-20 times higher than the transversal one, but carrying out the FD parameter evaluations for multiple string segments would add a significant computational load, likely without any audible advances. The elongation of the string can be expressed as [19]

$$l_{\text{dev}}(t) = \int_0^{l_{\text{nom}}} \sqrt{1 + (y_x(t, x))^2} dx - l_{\text{nom}}, \quad (7)$$

where l_{nom} is the nominal string length, x is the spatial coordinate along the string, and y is the displacement of the string. The first spatial derivative, y_x , suggests the use of slope waves in the elongation calculation, and thus equation (7) can be approximated for the digital waveguide as [28]

$$L_{\text{dev}}(n) = \sum_{m=0}^{\hat{L}_{\text{nom}}} \sqrt{1 + [s_r(n, m) + s_l(n, m)]^2} - \hat{L}_{\text{nom}}, \quad (8)$$

where $s_r(n, m)$ and $s_l(n, m)$ are the slope waves at time instant n and position m , propagating to the right and to the left, respectively. \hat{L}_{nom} is the rounded nominal string length. To reduce the computational complexity, equation (8) can be further simplified using a truncated Taylor series expansion to [28]

$$L_{\text{dev}}(n) \approx \frac{1}{2} \sum_{m=0}^{\hat{L}_{\text{nom}}} [s_r(n, m) + s_l(n, m)]^2, \quad (9)$$

while still maintaining a sufficient accuracy. The approximated delay variation of the total DWG can be obtained from equation (9) as [28]

$$D_{\text{dev}}(n) \approx -\frac{1}{2} \sum_{l=n-1-\hat{L}_{\text{nom}}}^{n-1} \left(1 + \frac{EA}{K_0}\right) \frac{L_{\text{dev}}(l)}{L_{\text{nom}}}, \quad (10)$$

where E is Young's modulus, A is the cross-sectional area of the string, and K_0 is the nominal tension corresponding to the string at rest. The length of the string in samples is denoted as $L_{\text{nom}} = l_{\text{nom}} f_s / c_{\text{nom}}$, where f_s is the temporal sampling frequency and c_{nom} is the nominal wave propagation speed.

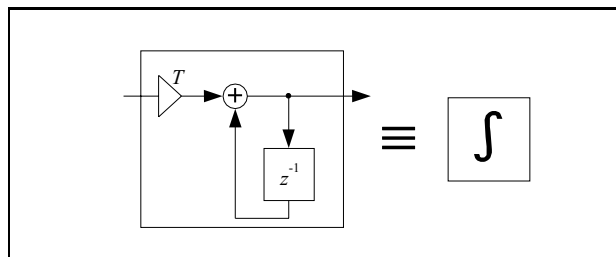


Figure 7. The integrator block is implemented by summing up consecutive samples.

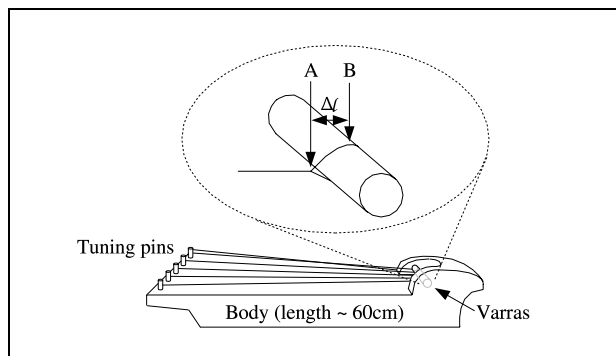


Figure 8. Illustration of the kantele. The string termination at varras is magnified for clarity. A denotes the termination point for vertical vibration of the string, while B denotes the termination point for horizontal vibration. Δl stands for the distance between A and B .

Since the system under consideration uses a distributed set of delay elements, the desired delay for each basic element is

$$d_{\text{partial}} = 1 + \frac{D_{\text{dev}}}{\hat{L}_{\text{nom}}}. \quad (11)$$

The coefficient a in equation (6) can now be expressed as [26, 24]

$$a = \frac{d_{\text{partial}} - 1}{d_{\text{partial}} + 1}, \quad (12)$$

where d_{partial} is the delay intended for a single allpass filter. Note that previous studies have used a different sign for a in equations (6) and (12), although the operation of the allpass filter remains the same.

4. Synthesis model of the kantele using nonlinear digital waveguides

In this section, we demonstrate the nonlinear DWG formulation by constructing a two-polarizational synthesis model of the kantele, a Finnish folk music instrument.

4.1. Acoustical analysis of the kantele

The kantele is a bridgeless plucked string instrument with usually five metal strings in its basic form (see Figure 8). The strings are terminated at one end by metal tuning pins,

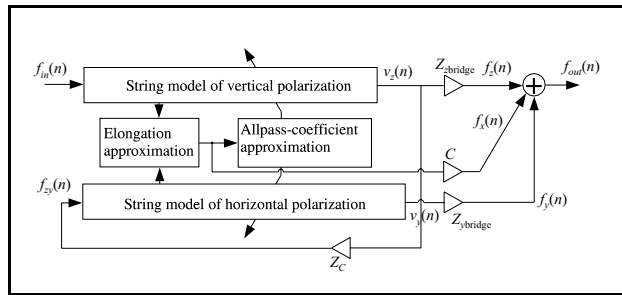


Figure 9. A kantele string model. The one-polarization string model blocks are identical to what is illustrated in Figure 5, but have different string lengths. It is important to note that the coupling between the vibrational polarizations in a real physical system is more complicated but a simplified one-way coupling is used here for ease of simulation.

which are screwed directly into the soundboard. At the opposite end all strings are wound once around a horizontal metal bar called the varras and then knotted. Because of the nonzero distance between the center of the varras and the knot, the vibrations in two polarization planes have different effective lengths; the varras is the termination point for horizontal vibration, while the knot acts as the termination for the vertical vibration, as illustrated in Figure 8. This phenomenon causes the total vibration of the string to have two fundamental components with slightly different frequencies, producing beating [29]. A more detailed structure and acoustical analysis of the kantele can be found in [29]. A study of the history of kantele and an acoustically improved new design are presented in [31].

4.2. A novel kantele string model

The novel synthesis model of a single kantele string is constructed using two single-polarization time-varying DWG models, illustrated in Figure 5, and connecting them together via a scaling coefficient for modeling the coupling between the two polarizations. We restrict the coupling to being one-directional in order to avoid stability problems which would otherwise rise due to the feedback loop formed from the interconnected strings, as suggested by Karjalainen et al. [32]. Clearly, the actual physical coupling is two-directional. The elongation approximations of the strings and the resulting allpass filter coefficient values are evaluated separately for the two DWG models using the arithmetics described in section 3.2. The structure of the novel kantele string model is illustrated in Figure 9. In this model, $v_y(n)$ and $v_z(n)$ represent the velocity signals coming from the strings vibrating vertically and horizontally, respectively.

It is important to note that while $v_y(n)$ and $v_z(n)$ can be obtained anywhere along the string, in this case they are evaluated at the termination points, so that terminal impedances can be used. $Z_{ybridge}$ and $Z_{zbridge}$ represent the vertical and horizontal terminal impedances, respectively. Z_c stands for the coupling impedance from vertical to horizontal string vibration polarization, and $f_{yz}(n)$ represents the corresponding driving force. The forces to

the termination caused by the two one-polarizational vibrations are denoted by $f_y(n)$ and $f_z(n)$. The connection from the elongation approximation block to the output simulates the direct coupling of the TM to the instrument body [20]. A scaling coefficient, denoted by C , is used to control the amount of this coupling. The output of the whole two-polarization string model is finally presented as a force signal $f_{out}(n)$ exerted to the string termination point. It must also be noted that this model simulates only a single kantele string, and a model of the instrument body must also be added, if realistic sound synthesis is desired.

A real-time sound synthesis model of a kantele is constructed using a block-based, efficient audio-DSP-tool, the BlockCompiler. The algorithm used is efficient enough to process a five-string kantele model on an ordinary laptop computer at a 44.1 kHz sampling rate. A detailed description of the BlockCompiler is presented by Karjalainen [33].

5. Finite difference approach

In the previous section we discussed string synthesis via discretizing the d'Alembert's solution to the 1-D wave equation. Another approach is to discretize the wave equation itself, for example by substituting finite difference terms for the derivatives in the wave equation (equation (1) in the case of an ideal string). This mode of operation is commonly known as the *finite difference method* (FDM), and it was first used for sound synthesis purposes by Hiller and Ruiz in the early seventies [6] and [34]. Finite differences had already earlier been used in mathematics for numerical solving of partial differential equations. A fine introduction to FDM in the synthesis of plucked string instruments can be found in [35]. Below, we follow the guidelines provided in [22] in deriving the FDM recurrence equation.

5.1. Ideal finite difference string

The partial derivatives in the 1-D wave equation (1) can be replaced by finite differences:¹

$$y_t(t, x) \approx \frac{y(t, x) - y(t-T, x)}{T} \tag{13}$$

and

$$y_x(t, x) \approx \frac{y(t, x) - y(t, x-X)}{X}. \tag{14}$$

Using the finite difference approximation for the second-order derivatives in the wave equation (1) leads to

$$\begin{aligned} & K \frac{y(t, x+X) - 2y(t, x) + y(t, x-X)}{X^2} \\ & = \epsilon \frac{y(t+T, x) - 2y(t, x) + y(t-T, x)}{T^2}. \end{aligned} \tag{15}$$

¹ It is important to note, that the finite difference scheme used in equations (13) and (14) was only chosen for simplicity, and other schemes could be used as well. For a discussion of using an implicit finite-difference scheme, see section 8.

Solving (15), we get

$$y(t+T, x) = \frac{KT^2}{\epsilon X^2} [y(t, x+X) - 2y(t, x) + y(t, x-X)] + 2y(t, x) - y(t-T, x). \quad (16)$$

Next, we define the relationship between the spatial and temporal sampling steps with [35]

$$r = c \frac{T}{X} \leq 1, \quad (17)$$

where the “less than unity”-restriction is called the Von Neumann stability condition. Using this together with the definition of transversal wave velocity (equation 2), equation (16) can be written as

$$y(t+T, x) = r^2 [y(t, x+X) + y(t, x-X)] + 2(1 - r^2)y(t, x) - y(t-T, x). \quad (18)$$

If we now do the discretization by denoting $t \rightarrow t(n) = nT$ and $x \rightarrow x(m) = mX$, as we did in section 3, we end up with the finite difference approximation [35]

$$y(n+1, m) \approx r^2 [y(n, m+1) + y(n, m-1)] + 2(1 - r^2)y(n, m) - y(n-1, m). \quad (19)$$

If we set $r = 1$, (19) becomes

$$y(n+1, m) = y(n, m+1) + y(n, m-1) - y(n-1, m), \quad (20)$$

which is the finite difference equation of an ideal string. The equality of equation (20) can be checked by substituting the waveguide decomposition (equation 5) in the right-hand side of equation (20) [22].

Since the length of the string must again have integer values, correct tuning of the string becomes difficult. It has been shown [35] that choosing $r < 1$ in equation (17) results in lowering the fundamental frequency of the string. Therefore, the finite difference string can be tuned via the parameter r .

Choosing $r < 1$ also gives rise to an unwanted numerical dispersion phenomenon called *grid dispersion* [7], where the wave velocity in the numerical implementation will be less than the ideal physical wave velocity. This artificial dispersion affects primarily the upper harmonics, where the frequencies will be underestimated. If a typical error of 1% in the generated frequencies is allowed, the difference between the tuning coefficient r and unity should not be greater than 10^{-4} [35]. If the constraints between the correct tuning and grid dispersion do not yield satisfactory results, the spatial density of the grid should be increased. This is known as *spatial oversampling*.

5.2. Boundary conditions and string excitation

Since the spatial coordinate m of the string must lie between 0 and \hat{L}_{nom} , problems arise near the ends of the string when evaluating equation (20) because spatial points outside the string are needed. The problem can

be solved by introducing boundary conditions that define how to evaluate the string movement when $m = 0$ or $m = \hat{L}_{\text{nom}}$. The simplest approach, introduced already in [6], would be to assume that the string terminations be rigid, so that $y(n, 0) = y(n, \hat{L}_{\text{nom}}) = 0$. This results in a phase-inverting termination, which suits perfectly the case of an ideal string. For other types of string termination, several models have been introduced (see e.g. [6], [35], and [36]). Generally, the nonrigid string terminations lead to frequency-dependent losses in the string model.

For the FDM string excitation, a useful method has been proposed in [36]. It is conceptually simple and allows for interaction with the string during run-time. There,

$$y(n, m) \leftarrow y(n, m) + \frac{1}{2}u(n) \quad (21)$$

and

$$y(n, m+1) \leftarrow y(n, m+1) + \frac{1}{2}u(n) \quad (22)$$

are inserted into the string, which causes a “boxcar” block function to spread in both directions from the excitation point pair. The wave component $u(n)$ is now used as the excitation signal in a similar way as the exciting force signal $F(n)$ in section 3.2.

5.3. Finite difference approximation of a lossy string

Frequency-independent losses can be modeled in an FDM string by discretizing the velocity-dependent damping term in the lossy 1D wave equation (3). This results in two additional scaling coefficients in the recurrence equation [35]

$$y(n+1, m) = p(y(n, m+1) + y(n, m-1)) - qy(n-1, m), \quad (23)$$

where the values of p and q determine the amount of losses. Generally, p and q may depend on the spatial index m , but since homogeneous strings are considered here, this dependency is omitted. Values

$$q = p^2, \quad |p| \leq 1 \quad (24)$$

ensure the stability of a linear finite difference string with frequency-independent losses [37]. Note that the sign difference of p and q in [37] has already been taken care of in equation (23). Modeling of frequency-dependent losses by discretizing the lossy wave equation leads to an implicit recurrence equation, which can be evaluated, if suitable approximations are made [35].

6. Nonlinear finite difference string

Implementing tension modulation in a digital waveguide string in section 3 was not an overly difficult task. This was due to the fact that the implementation of a DWG string is essentially a feedback loop with delay, and therefore modulating the delay time of this loop corresponded to modulating the wave velocities. In FDM strings, however, such

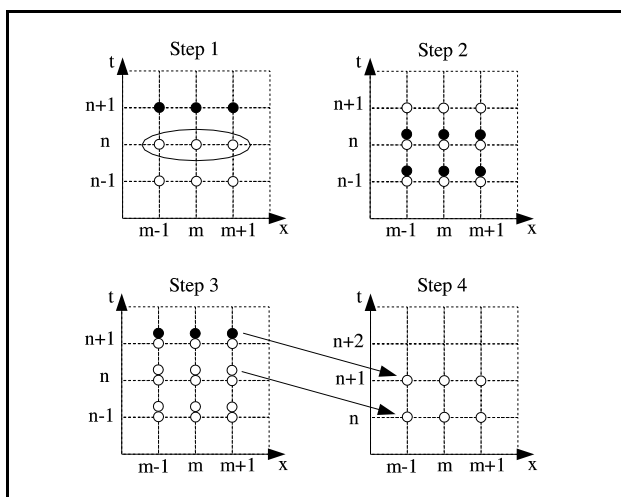


Figure 10. Illustration of the nonlinear FD algorithm on a spatio-temporal grid. The vertical axis denotes the time, while the horizontal axis denotes the spatial location on the string. The illustration is shown only for a string segment of length $N = 3$, for clarity. In each step of the algorithm, most recently evaluated values are presented as black dots, while earlier values are presented as white dots.

an approach would not lead to satisfactory results, since the physical quantities (e.g. displacement) themselves are present in the string model, and not their wave decompositions².

Instead, we concluded that in order to correctly model the TM in a FDM string, we first have to evaluate the recurrence equation and use these three snapshots of the string (at time instants $n+1$, n , and $n-1$) in interpolating two new string states at time instants $n + \Delta$ and $n + \Delta - 1$, where $0 < \Delta < 1$. Using these two string states we then evaluate the recurrence equation, in order to obtain the string state at time $n + 1 + \Delta$. It is important to note that the interpolation here is in effect stretching the time axis so that the wave propagation velocities are altered, whereas in the DWG model the allpass filters perform the interpolation in the spatial domain.

This algorithm can also be seen as using two FDM systems in implementing the nonlinear string. The elongation of the string would be evaluated from one system, and the result, the stretched string state, would be updated to the other system. Figure 10 illustrates this procedure on the spatio-temporal grid.

In step 1, the two initial states have been assigned for the string, and the state at the next instant (in the linear case) is obtained by the standard recurrence equation (20). The grid values which represent the state of the string at the corresponding time instant are circled in step 1. In step 2, sample values corresponding to the TM have been interpolated from the string states in step 1. In step 3, equation (20) has been applied on the values evaluated in step 2, in

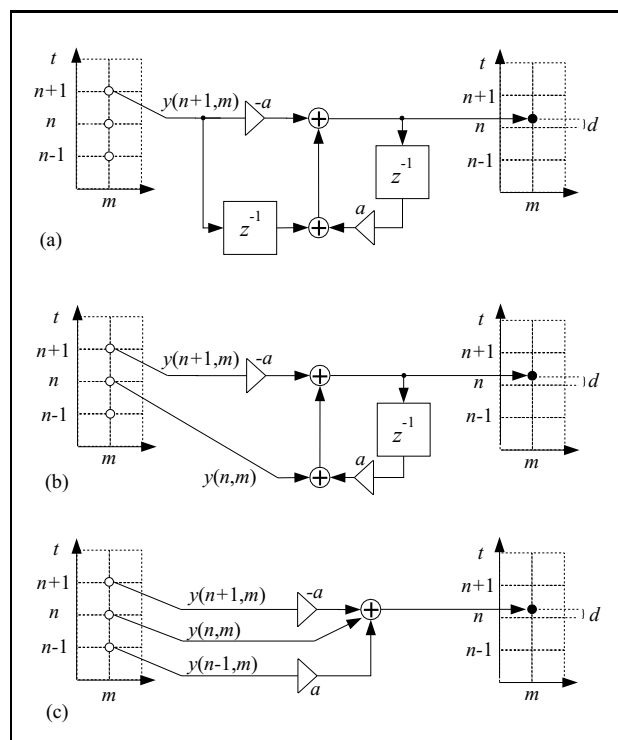


Figure 11. Illustration of the interpolation process due to the change in the string’s length. The spatio-temporal grids on the left and right represent the linear and interpolated string states, respectively. The fractional delay value caused by the interpolation is denoted by d . The interpolation process in (a) is simplified in (b), and further in (c).

order to obtain the string state corresponding to the change in tension. The two most recently obtained states are now taken as the “initial states” in step 4, and we can return to step 1.

As seen in Figure 10, the tension modulation corresponds here to interpolating the string state in the temporal domain. The elongation of the FDM string was evaluated similarly to what was done in equation (9), except that here the slope of the string was obtained by taking the difference of the displacements between two adjacent string segments, rather than summing up the slope wave components. In the following, we will have a closer look at the interpolation process.

6.1. String state interpolation

We chose again to use first-order allpass filters in interpolating the string state from the linear model (step 2 in Figure 10). Figure 11(a) illustrates how the interpolated value of $y(n, m)$ is obtained from the linear values. The spatio-temporal grid on the left represents the string state in the linear case, while the spatio-temporal grid on the right represents the string state after spatial interpolation. The structure between the two grids is the block diagram of a first-order allpass filter (equation 6). The coefficient a for the allpass filter was evaluated as presented earlier by equations (9)–(12).

² Such a system which deals with the physical quantities themselves is called a *Kirchhoff* model, as opposed to a *wave* model, which deals with the wave components of the physical quantities.

In this figure, we notice that the allpass filter uses the value of $y(n+1, m)$ delayed by one sample, thus corresponding to $y(n, m)$. Clearly, this can be obtained directly from the grid on the left, and the branch on the left containing the unit delay can be reformed. The result is shown in Figure 11(b). Here we also note that the interpolation system uses its own output at the previous time instant. This is actually the same as using the value of $y(n-1, m)$, because it is the same as the output of the interpolation process one time step ago (this might be best understood by noting that the bottom row of step 4 in Figure 10 is the same as the bottom row of step 1 at the next time instant). Thus, Figure 11(b) can be further simplified to Figure 11(c).

Having this said, the recurrence equation for the time-varying finite difference string with frequency-independent damping can be written as

$$y_1(n+1, m) = p(y_2(n, m-1) + y_2(n, m+1)) - qy_2(n-1, m), \quad (25)$$

where

$$\begin{aligned} y_2(n, m-1) &= -ay_1(n+1, m-1) + y_1(n, m-1) \\ &\quad + ay_1(n-1, m-1), \\ y_2(n, m+1) &= -ay_1(n+1, m+1) + y_1(n, m+1) \\ &\quad + ay_1(n-1, m+1), \\ y_2(n-1, m) &= -ay_1(n, m) + y_1(n-1, m) \\ &\quad + ay_1(n-2, m). \end{aligned}$$

Here the coefficients p and q incorporate the frequency-independent losses and y_1 and y_2 refer to the linear and interpolated strings, respectively. Simplifying and rearranging we end up with an equation containing only terms of y_1 , and the subscript may therefore be omitted

$$\begin{aligned} y(n+1, m) &= -pay(n+1, m-1) - pay(n+1, m+1) \\ &\quad + py(n, m+1) + qay(n, m) + py(n, m-1) \\ &\quad + pay(n-1, m+1) - qy(n-1, m) \\ &\quad + pay(n-1, m-1) - qay(n-2, m). \end{aligned} \quad (26)$$

This equation is illustrated with a block diagram in Figure 12 along with its abstraction. A nonlinear FDM string can be constructed by connecting several of these blocks together and using the string elongation in controlling the amount of interpolation. We will refer to such a block as a time-varying *finite difference time-domain (FDTD) element*. Illustration of the lossless time-varying FDTD element can be found in Figure 13, where p and q equal unity and have therefore been left out.

6.2. String excitation and termination

For the interaction with the time-varying FDTD string model, we chose to use the “boxcar” excitation model discussed in section 5.2, so that the excitation signal could again be interpreted as a force signal. Figure 14 presents an interaction block to be used with a time-varying FDTD string. We will call such a block the *FDTD interaction element*.

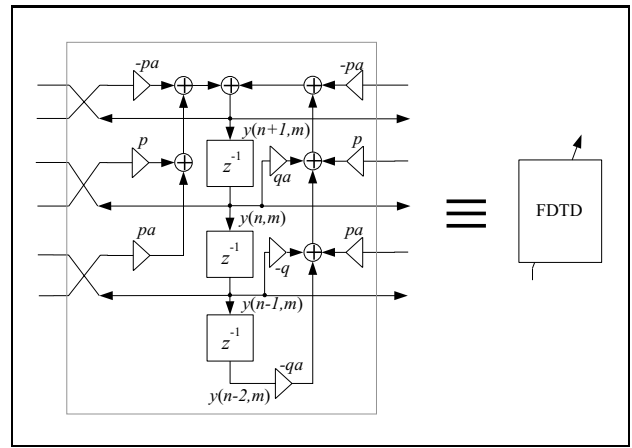


Figure 12. Illustration of the time-varying FDTD element together with its abstraction. A lossless time-varying FDTD element can be found in Figure 13.

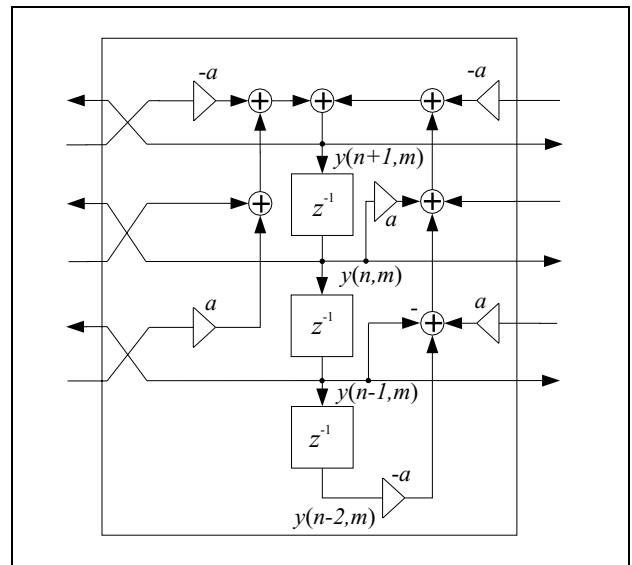


Figure 13. Illustration of the lossless time-varying FDTD element.

Using these DSP blocks, we can construct a one-polarization *nonlinear FDTD (NFDTD) string*, as illustrated in Figure 15.

We chose to use rigid terminations for our nonlinear finite difference string model, since the modeling of frequency-dependent losses is not a key aspect of this study. Fixed terminations do not ruin the generation of missing harmonics in our model either, since the TMDF coupling is implemented in a different manner, as explained below.

6.3. NFDTD string with generation of missing harmonics

In order to model the generation of missing harmonics in a NFDTD string, we constructed a model, where an additional interaction element is placed between the last FDTD element and the termination for feeding the TMDF to the string. Since the spatial distance between the last FDTD

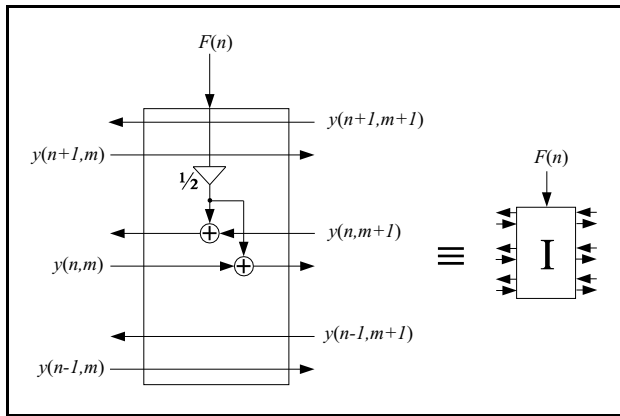


Figure 14. Illustration of the FDTD interaction element together with its abstraction. The excitation algorithm is defined by equations (21) and (22).

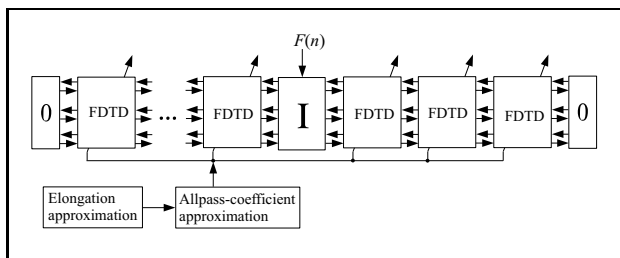


Figure 15. One-polarizational NFDTD string. The string consists of the time-varying FDTD elements illustrated in Figure 12. The zero-blocks at the terminations give zero as an output regardless of the input values, thus implying a rigid termination. The excitation to the string can be inserted as a force signal using a FDTD interaction element, illustrated in Figure 14.

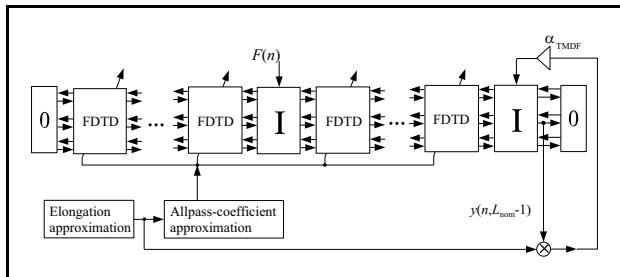


Figure 16. Illustration of the NFDTD string with a generation mechanism for missing harmonics. A second interaction element is added in order to feed the TMDF into the string. The scaling coefficient α_{TMDF} controls the amplitude of the missing harmonics. The string elongation is approximated from the displacements of each FDTD element.

element and the rigid termination is one sample, the vertical component of the TMDF can be seen to be equal to the product of the displacement of the last FDTD element and the tension. Here, we can replace the tension signal by the elongation signal, and introduce a scaling coefficient α_{TMDF} to control the amount of TMDF to be inserted to the interaction element at the termination. This is illustrated in Figure 16. The generation of missing har-

monics in a NFDTD model will be further discussed in the following section.

7. Simulation results

In this section we present the results obtained from the two nonlinear string algorithms, discussed in sections 3 and 5. The synthesis results are compared by simulating the same phenomena, namely the initial pitch glide and the generation of missing harmonics, using the two models. Stability issues and computational cost of the synthesis models are also discussed.

7.1. Synthesis results

The synthesis results reveal that both the nonlinear DWG and NFDTD models are able to realistically model the initial pitch glide phenomenon. Figure 17 illustrates the fundamental frequency behavior of a recorded kantele tone and the two synthesized tones. Here, the horizontal dotted line approximates the mean value of perceptual detection threshold of an initial pitch glide. The psychoacoustic detection threshold in the frequency region of these tones is about 5.4 Hz [38]. This shows that the fundamental frequency glide is an audible phenomenon in plucked string instruments such as the kantele even at modest plucking amplitudes, and thus it must be included in a synthesis model if realistic tones are desired.

The nonlinear DWG model used in this figure has a total delay line length of 55.125 samples and the allpass coefficient a is scaled using a constant value of 0.9 in order to correctly simulate the behavior of the recorded sample. The NFDTD string consists of 56 FDTD elements, and the fine-tuning parameter (a.k.a. Courant number, equation 17) has a value of $r = 0.998$. The allpass coefficient a is scaled using a coefficient 0.25 in the NFDTD case.

The modeling of the generation of missing harmonics can be implemented similarly in the distributed nonlinear DWG model as was suggested in [28]. If the boxcar integration of equation (10) is replaced with a leaky integrator having the transfer function

$$I(z) = g_p \frac{1 + a_p}{1 + a_p z^{-1}}, \tag{27}$$

the generation of the missing harmonics can be controlled via the integration parameter a_p . The variable g_p defines the gain of the integration.

Figure 18 shows the amplitude envelopes of the first three harmonics of a synthesized tone with two different a_p parameter values. The string was plucked close to 1/3rd of its length, and as can be seen in the figure, the missing harmonic in (a) has a gradual increase after the beginning transient, after which it experiences an exponential decay like all other modes.

It is worthwhile to note that the generation of missing harmonics in the nonlinear DWG model results from the properties of the integration of the elongation approximation, and is therefore not a physically justified process. Basically, here the integration error in the leaky integrator is

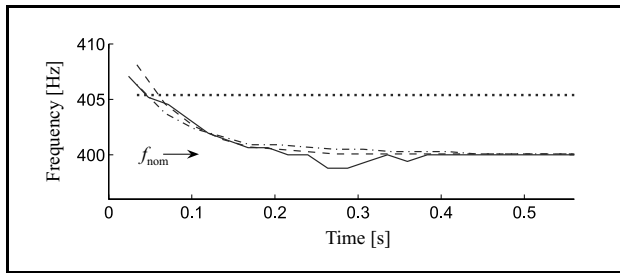


Figure 17. Fundamental frequencies as a function of time for a moderately-plucked recorded kantele tone (solid line), a synthesized nonlinear DWG tone (dashed line), and a synthesized NFDTD tone (dash-dotted line). The f_{nom} stands for the nominal fundamental frequency of the string, and the horizontal dotted line denotes the approximated detection threshold of a pitch drift ($f_{nom} + 5.4$ Hz), which suggests that the fundamental frequency drifts in all cases are audible.

responsible for feeding energy to the missing harmonics. Also, unlike the real physical phenomenon, the generation of missing harmonics in the nonlinear DWG case does not depend on the rigidity of the terminations. Nevertheless, this feature can be exploited in emulating the real string behavior, when the integration parameters are properly adjusted. Details on tuning the leaky integrator parameters can be found in [28].

Modeling the generation of missing harmonics in a NFDTD string is however not so simple. Even if a leaky integrator is used in the elongation calculation, its parameters do not have a desirable effect on the missing harmonics. This does not seem too surprising when considering the major differences of these two algorithms and it is the reason that forced us to use an alternative mechanism for creating the missing harmonics in the previous section (Figure 16).

Figure 19 represents the behavior of the first three harmonics of a tone synthesized by this model. It can be seen that the missing harmonics can be “lifted” by choosing a proper value for α_{TMDF} . The stability of the system however poses an upper limit for the α_{TMDF} coefficient, since the TMDF mechanism continuously feeds energy to the string. According to our experience, generating missing harmonics with amplitudes greater than what is shown in Figure 19 is difficult.

7.2. Stability issues and computational comparison

We found the nonlinear DWG algorithm to remain stable for nearly all parameter and excitation values. Only highly exaggerated nonlinearity scaling values together with high excitation impulses resulted in stability problems. We thus conclude that the nonlinear DWG waveguide has no real stability problems when synthesis of natural plucked-instrument sounds are desired.

We studied the stability of the NFDTD algorithm using the Von Neumann analysis [39] in the time-invariant case, i.e. parameter a of equation (26) was kept constant. The basic idea of this method is to calculate the spatial Fourier spectrum of the system under discussion at two consecutive time steps. An amplification function λ , which

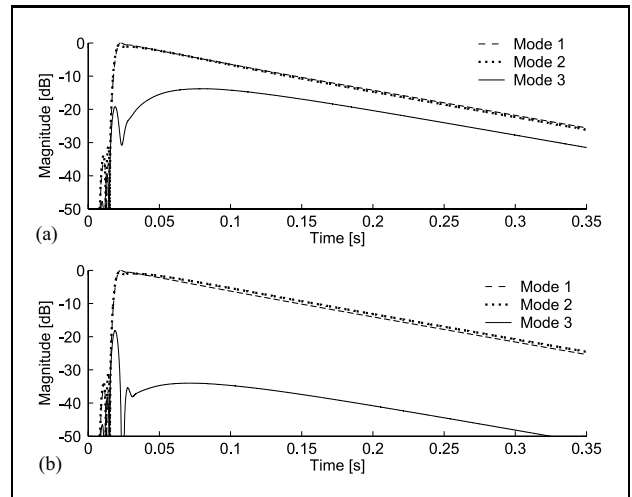


Figure 18. Generation of the missing harmonics in the nonlinear DWG model can be controlled via the leaky integrator parameters. Here, the string was plucked approximately at 1/3rd of its length, so every 3rd harmonic should be missing from the resulting spectrum. In a), $a_p = -0.2902$ and the third harmonic clearly rises after the initial transient. In b), $a_p = -0.9672$ and the third harmonic is more attenuated.

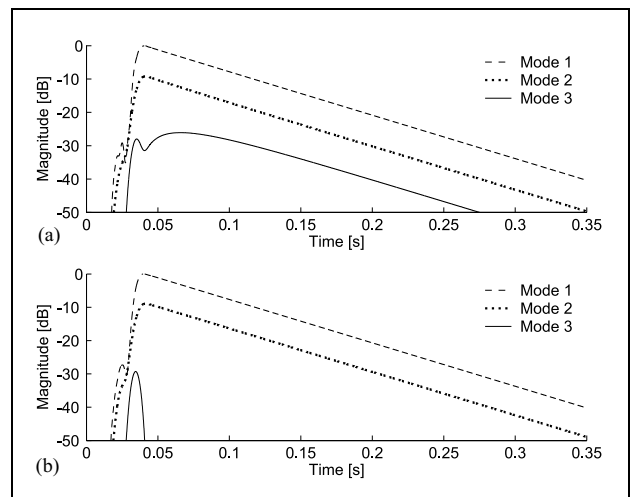


Figure 19. Generation of missing harmonics in a NFDTD string. The string was plucked again approximately at 1/3rd of its length, and the coupling of the TMDF to the transversal vibration was controlled using a scaling coefficient α_{TMDF} . In a), the scaling coefficient has a value of $\alpha_{TMDF} = 4$, and the missing third harmonic can be seen rising after the initial transient. In b), $\alpha_{TMDF} = 0$, and generation of missing harmonics does not take place.

shows how the spatial spectrum evolves with time, can then be derived from the two spectra. If the absolute value of this amplification function remains below unity, stability is guaranteed. Formally, the Von Neumann analysis for the NFDTD algorithm goes as follows [4]:

If the spatial inverse Fourier transform is defined as

$$y(n, m) = \mathcal{F}\{Y(n, \xi)\}^{-1} = \lambda^n e^{im\theta}, \quad (28)$$

where θ is the spatial frequency and i is the imaginary unit, the nonlinear finite-difference recurrence equation (26) can be written as

$$\begin{aligned} \lambda^{n+1} e^{im\theta} &= -pa\lambda^{n+1} e^{i(m+1)\theta} - pa\lambda^{n+1} e^{i(m-1)\theta} \\ &+ p\lambda^n e^{i(m+1)\theta} + qa\lambda^n e^{im\theta} + p\lambda^n e^{i(m-1)\theta} \\ &+ pa\lambda^{n-1} e^{i(m+1)\theta} - q\lambda^{n-1} e^{im\theta} \\ &+ pa\lambda^{n-1} e^{i(m-1)\theta} - qa\lambda^{n-2} e^{im\theta}. \end{aligned} \quad (29)$$

Dividing with $\lambda^{n-2} e^{im\theta}$ and rearranging, we have

$$\begin{aligned} (1 + pa e^{i\theta} + pa e^{-i\theta}) \lambda^3 \\ + (-p e^{i\theta} - qa - p e^{-i\theta}) \lambda^2 \\ + (-pa e^{i\theta} + q - pa e^{-i\theta}) \lambda - qa = 0. \end{aligned} \quad (30)$$

Using the Euler's equation leads to a simpler form

$$A\lambda^3 + B\lambda^2 + C\lambda + D = 0, \quad (31)$$

where

$$\begin{aligned} A &= 1 + 2pa \cos(\theta), \\ B &= -qa - 2p \cos(\theta), \\ C &= q - 2pa \cos(\theta), \\ D &= -qa. \end{aligned} \quad (32)$$

In order to get the amplification function λ , we would now have to solve the third-order equation (31). Unfortunately, the solution of this equation is complicated and involves dozens of terms. If we want to consider the stability of the lossless NFDTD string, we can substitute $p = q = 1$. This simplifies the solution of equation (31) enough to enable numerical stability analysis for the amplification function. The absolute value of the amplification function λ is illustrated in Figure 20 as a function of the interpolation coefficient a , and the spatial frequency θ .

It is important to note that this stability analysis is conducted on a *lossless* NFDTD string with *constant interpolation coefficient*. We can thus call this system time-invariant (normally the interpolation coefficient depends on the string elongation).

Figure 20 reveals that in the lossless case, the time-invariant version of the NFDTD algorithm is unstable for all but very small a parameter values. Making the algorithm time-variant results in an even more unstable system. In a practical lossy string implementation, however, the NFDTD string remained stable for normal excitation amplitudes (i.e. excitation amplitudes commonly used when playing real string instruments).

The computational complexities of the two algorithms are different. Since the models consist mainly of the basic string blocks (basic elements in the DWG case and FDTD elements in the finite difference case), the differences in the computation of the basic string blocks dominate the computational needs of the algorithms.

The basic element (Figure 4) consists of four multiplications and two summations per time sample, whereas the FDTD element (Figure 12) requires a total of nine multiplications and eight summations for computing one time

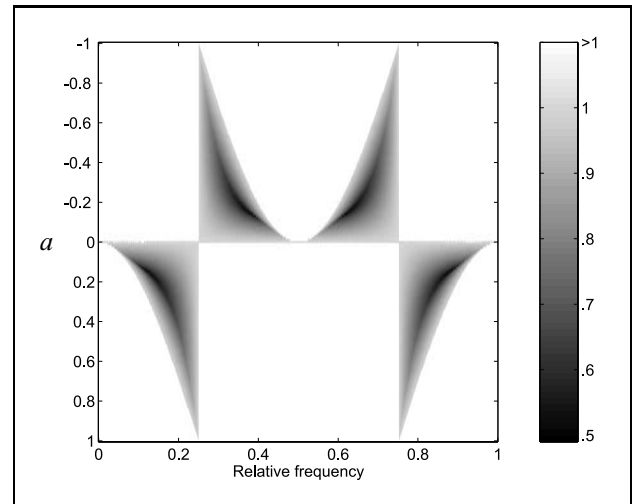


Figure 20. Absolute value of amplification function of a NFDTD algorithm. The white color denotes areas where the amplification function exceeds unity, i.e. when the model becomes unstable.

sample. Although the interaction and termination blocks are much simpler in the finite difference case, the typically large number of the string elements turns the favor to the nonlinear DWG model. If the computational cost of the string elongation approximation is taken into account, the NFDTD algorithm can be seen to have twice the computational complexity of its digital waveguide counterpart. For a more thorough comparison of the two presented algorithms, see [4].

8. Conclusions and future work

Two algorithms for modeling spatially distributed nonlinear strings in a physically meaningful way were presented: a nonlinear digital waveguide algorithm and a nonlinear finite difference algorithm. The former uses first-order allpass filters distributed along a delay line for modulating the total delay of the string loop, while the latter one uses first-order allpass filters for interpolating between time samples in the linear recurrence equation. Both techniques evaluate the control signals for the allpass filters from the elongation of the string. The amount of nonlinearity, among with other physical parameters, can be adjusted in both string models. A physical model of a kantele string was presented using the nonlinear digital waveguide string algorithm.

Realistic simulation of the initial pitch glide phenomenon can be performed with both algorithms, but modeling of the generation of missing harmonics can be realistically obtained only using the nonlinear digital waveguide model, due to stability problems of the nonlinear finite difference algorithm. Computational complexities of the two algorithms were also compared.

As stated in section 5.1, the explicit finite difference scheme was chosen for simplicity. Another option would be to use an implicit scheme, such as a θ scheme [40], where the temporal and spatial derivatives of the wave equation (equation 1) are averaged in space and time, re-

spectively. Using such a scheme would lead to an unconditionally stable finite difference algorithm and thus liberate us from the Von Neumann stability condition (equation 17). The implicit form of this scheme would however call for a matrix formulation instead of a simple recurrence equation, and probably increase the computational load of the algorithm. Construction of such an algorithm is left for future work.

Acknowledgement

Thanks to Dr. Cumhuri Erkut and Dr. Lutz Trautmann for suggestions and discussions. This work was supported by the ALMA project (IST-2001-33059), the Academy of Finland project SA 104934, and the Helsinki Graduate School of Electrical and Communications Engineering.

References

- [1] M. Karjalainen, C. Erkut: Digital waveguides vs. finite difference schemes: Equivalence and mixed modeling. *EURASIP Journal on Applied Signal Processing* (June 2004) 978–989. Special issue on Model-Based Sound Synthesis.
- [2] C. Erkut, M. Karjalainen: Finite difference method vs. digital waveguide method in string instrument modeling and synthesis. *Proceedings of the International Symposium on Musical Acoustics (ISMA 2002)*, Mexico City, Mexico, December 9-13, 2002.
- [3] J. Pakarinen, M. Karjalainen, V. Välimäki: Modeling and real-time synthesis of the kantele using distributed tension modulation. *Proc. Stockholm Music Acoustics Conference*, Stockholm, Sweden, August 6-9, 2003, 409–412.
- [4] J. Pakarinen: Spatially distributed computational modeling of a nonlinear vibrating string. Diploma Thesis. Helsinki University of Technology, June 14, 2004. Available on-line at <http://www.acoustics.hut.fi/publications/>.
- [5] N. H. Fletcher, T. D. Rossing: *The physics of musical instruments*. Springer-Verlag, New York, USA, 1988.
- [6] L. Hiller, P. Ruiz: Synthesizing musical sounds by solving the wave equation for vibrating objects: Part I. *Journal of the Audio Engineering Society* **19** (June 1971) 462–470.
- [7] A. Chaigne, A. Askenfelt: Numerical simulations of piano strings. I. A physical model for a struck string using finite difference methods. *Journal of the Acoustical Society of America* **95** (February 1994) 1112–1118.
- [8] M. Podlesak, A. Lee: Dispersion of waves in piano strings. *Journal of the Acoustical Society of America* **83** (1988) 305–317.
- [9] D. Hall: Piano string excitation in the case of small hammer mass. *Journal of the Acoustical Society of America* **79** (1986) 141–147.
- [10] D. Hall: Piano string excitation II: General solution for a hard narrow hammer. *Journal of the Acoustical Society of America* **81** (1987) 535–546.
- [11] D. Hall: Piano string excitation III: General solution for a soft narrow hammer. *Journal of the Acoustical Society of America* **81** (1987) 547–555.
- [12] H. Suzuki: Model analysis of a hammer-string interaction. *Journal of the Acoustical Society of America* **82** (1987) 1145–1151.
- [13] X. Boutillon: Model for piano hammers: Experimental determination and digital simulation. *Journal of the Acoustical Society of America* **83** (1988) 746–754.
- [14] M. E. McIntyre, J. Woodhouse: On the fundamentals of bowed string dynamics. *Acustica* **43** (1979) 93–108.
- [15] J. Woodhouse: Idealised models of a bowed string. *Acustica* **79** (1993) 233–250.
- [16] L. Cremer: *The physics of the violin*. MIT Press, Cambridge, MA, 1983.
- [17] H. A. Conklin: Generation of partials due to nonlinear mixing in a stringed instrument. *Journal of the Acoustical Society of America* **105** (January 1999) 536–545.
- [18] B. Bank, L. Sujbert: Modeling the longitudinal vibration of piano strings. *Proc. Stockholm Music Acoustics Conference*, Stockholm, Sweden, August 6-9, 2003, 143–146.
- [19] K. A. Legge, N. H. Fletcher: Nonlinear generation of missing modes on a vibrating string. *Journal of the Acoustical Society of America* **76** (July 1984) 5–12.
- [20] T. Tolonen, C. Erkut, V. Välimäki, M. Karjalainen: Simulation of plucked strings exhibiting tension modulation driving force. *Proceedings of the International Computer Music Conference*, Beijing, China, October 22-28, 1999, 5–8.
- [21] K. Karplus, A. Strong: Digital synthesis of plucked-string and drum timbres. *Computer Music Journal* **7** (1983) 43–55.
- [22] J. O. Smith: Principles of digital waveguide models of musical instruments. *Applications of Digital Signal Processing to Audio and Acoustics*, (M. Kahrs and K. Brandenburg, eds.) (February 1998) 417–466.
- [23] J. O. Smith: Physical modeling using digital waveguides. *Computer Music Journal* **16** (Winter 1992) 74–87.
- [24] T. I. Laakso, V. Välimäki, M. Karjalainen, U. K. Laine: Splitting the unit delay - tools for fractional delay filter design. *IEEE Signal Processing Magazine* **13** (1996) 30–60.
- [25] V. Välimäki, T. I. Laakso: Principles of fractional delay filters. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Istanbul, Turkey, 5-9 June 2000, 3870–3873.
- [26] V. Välimäki: Discrete-time modeling of acoustic tubes using fractional delay filters. Doctoral dissertation, Helsinki Univ. of Technol., Acoustics Lab Report Series, Report no. 37, 1995. Available on-line at <http://www.acoustics.hut.fi/publications/>.
- [27] V. Välimäki, T. Tolonen, M. Karjalainen: Plucked-string synthesis algorithms with tension modulation nonlinearity. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Phoenix, Arizona, March 15-19, 1999, 977–980.
- [28] T. Tolonen, V. Välimäki, M. Karjalainen: Modeling of tension modulation nonlinearity in plucked strings. *IEEE Transactions on Speech and Audio Processing* **8** (May 2000) 300–310.
- [29] C. Erkut, M. Karjalainen, P. Huang, V. Välimäki: Acoustical analysis and model-based sound synthesis of the kantele. *Journal of the Acoustical Society of America* **112** (October 2002) 1681–1691.
- [30] J. R. Pierce, S. A. Van Duyne: A passive nonlinear digital filter design which facilitates physics-based sound synthesis of highly nonlinear musical instruments. *Journal of the Acoustical Society of America* **101** (February 1997) 1120–1126.
- [31] J. Pölkki, C. Erkut, H. Penttinen, M. Karjalainen, V. Välimäki: New designs for the kantele with improved sound radiation. *Proc. Stockholm Music Acoustics Conference*, Stockholm, Sweden, August 6-9, 2003, 133–136.

- [32] M. Karjalainen, V. Välimäki, T. Tolonen: Plucked-string models: From the Karplus-Strong algorithm to digital waveguides and beyond. *Computer Music Journal* **22** (1998) 17–32.
- [33] M. Karjalainen: BlockCompiler: Efficient simulation of acoustic and audio systems. Proc. 114th AES Convention, Amsterdam, The Netherlands, 22-25 March 2003.
- [34] L. Hiller, P. Ruiz: Synthesizing musical sounds by solving the wave equation for vibrating objects: Part II. *Journal of the Audio Engineering Society* **19** (June 1971) 542–551.
- [35] A. Chaigne: On the use of finite differences for musical synthesis. Application to plucked stringed instruments. *Journal d'Acoustique* **5** (1992) 181–211.
- [36] M. Karjalainen: 1-D digital waveguide modeling for improved sound synthesis. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Orlando, Florida, USA, May 13-17, 2002, 1869–1872.
- [37] C. Erkut, M. Karjalainen: Virtual strings based on a 1-D FDTD waveguide model: Stability, losses, and traveling waves. Proceedings of the Audio Engineering Society 22nd International Conference, Espoo, Finland, June 15-17, 2002, 317–323.
- [38] H. Järveläinen, V. Välimäki: Audibility of initial pitch glides in string instrument sounds. Proceedings of the International Computer Music Conference, Havana, Cuba, 17-23 September 2001, 282–285. Available on-line at <http://lib.hut.fi/Diss/2003/isbn9512263149/article3.pdf>.
- [39] J. C. Strikwerda: Finite difference schemes and partial differential equations. Wadsworth, Brooks & Cole, California, USA, 1989.
- [40] A. Chaigne, V. Doutaut: Numerical simulations of xylophones. I. Time-domain modeling of the vibrating bars. *Journal of the Acoustical Society of America* **101** (January 1997) 539–557.